OFFICE OF NAVAL RESEARCH

Fast Algorithms for Three-Dimensional Inverse Scattering Problems

ONR Young Investigator Award, 1990

Andrew E. Yagle

FINAL TECHNICAL REPORT

Grant #N00014-90-J-1897

September 30, 1994

Mathematical and Physical Sciences Directorate

Mathematical Sciences Division

Applied Analysis Program

Principal Investigator:

Andrew E. Yagle, Associate Professor

Department of Electrical Engineering and Computer Science

The University of Michigan, Ann Arbor, MI 48109-2122

Phone: (313) 763-9810. **Fax:** (313) 763-1503. E-mail: aey@eecs.umich.edu

THE UNIVERSITY OF MICHIGAN

Department of Electrical Engineering and Computer Science
Ann Arbor, Michigan 48109-2122
USA

19950420 001

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

REPORT DOCUMENTATION PAGE

Form Approved
OMB No 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	September 30, 1994	FINAL TECHNICAL REPORT

4. TITLE AND SUBTITLE
Fast Algorithms for Three-Dimensional Inverse Scattering Problems

5. FUNDING NUMBERS
N00014-90-J-1897

6. AUTHOR(S)
Andrew E. Yagle

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Dept. of Elec. Engin. and Comp. Sci.
The University of Michigan
Ann Arbor, MI 48109-2122

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Office of Naval Research
Mathematical Sciences Division
Code 1111
Arlington, VA 22217-5000

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE
Approved for public release; distribution unlimited.

13. ABSTRACT *(Maximum 200 words)*

This report is the FINAL report for grant N00014-90-J-1897. This project develops fast algorithms for multidimensional inverse scattering problems.

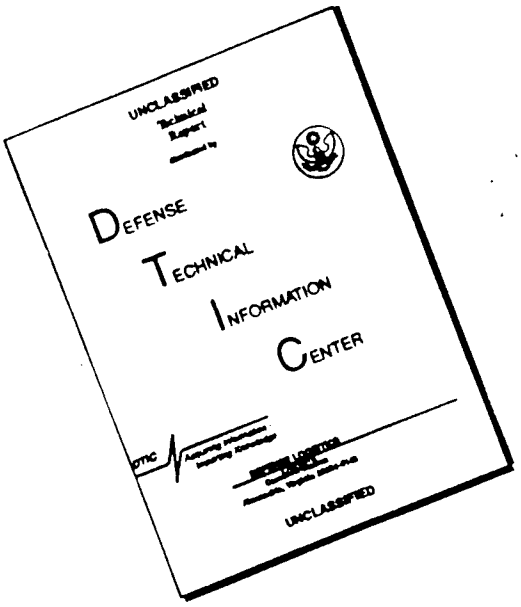
14. SUBJECT TERMS
Inverse scattering
Fast algorithms
Layer stripping algorithms

15. NUMBER OF PAGES

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	

DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

TABLE OF CONTENTS

I. SUMMARY OF RESEARCH OBJECTIVES 1

A. The Basic Problem

B. Approach to Solving Problem

C. Original Research Objectives

D. Additional Research Objectives

II. SUMMARY OF RESEARCH ACCOMPLISHMENTS 5

A. Original Research Objectives

B. Additional Research Objectives

C. Data Admissibility

D. Multiresolution Methods for the Inverse Radon Transform

III. PUBLICATIONS SUPPORTED BY GRANT #N00014-90-J-1897 11

IV. SUMMARIES OF APPENDICES 16

APPENDICES A-T 25

Accession For

NTIS GRA&I ☑

DTIC TAB ☐

Unannounced ☐

Justification

By

Distribution/

Availability Codes

Dist | Avail and/or Special

A-1

I. SUMMARY OF RESEARCH OBJECTIVES

A. The Basic Problem

The basic problem considered in this research is the inverse potential problem of reconstructing the scattering potential of a multidimensional Schrodinger equation from impulse reflection response data of several different types. This problem has obvious applications in: (1) exploration seismology, in which the potential represents the inhomogeneity of a scattering medium which is to be reconstructed; (2) nondestructive testing, which can be formulated similarly to exploration seismology, save for the availability of more scattering data; (3) inverse resistivity reconstruction, in which the potential is related to the electrical resistivity of the medium and the excitation is DC injected current; and (4) linear least-squares estimation of random fields. More details on these problems, including their mathematical formulation, can be found in Appendix A and the references cited therein.

The particular formulation addressed is the inverse scattering problem of reconstructing the scattering potential of a two-dimensional scattering medium from its backscattered reflection response to a planar impulsive wave normally incident on the medium from above. This formulation is obviously directly related to the applications noted above, since the planar impulsive wave can be generated by explosive sources on the surface of the medium (e.g., the earth's surface), and the response measured by geophones there. However, this formulation is also applicable to inverse problems that at first glance may seem totally unrelated to these. For example, the problem of reconstructing a scattering medium from its sinusoidal steady-state response to a set of point **harmonic** sources (i.e., single-frequency oscillators) can be formulated as this problem, even though the source and measurements are completely different (see Appendix A).

B. Approach to Solving Problem

There are integral equation based methods for solving the multidimensional Schrodinger equation inverse potential problem. These methods include the generalized Marchenko and Gel'fand-Levitan integral equations of Newton. However, these methods are numerically

untested, and would require a huge amount of computation.

This project pursued an alternative approach–**layer stripping**. Layer stripping algorithms recursively and differentially reconstruct the scattering potential, slice by slice, and simultaneously propagate the scattered wave field deeper into the scattering medium. They require much less computation than integral equation based methods, since they exploit time casuality and the Hankel structure in the integral equations. They are recursive in one spatial dimension, but parallelizable in time and all other spatial dimensions.

Layer stripping algorithms mathematically mimic the actual physical scattering process. They operate by recursively reconstructing the scattering potential at the wave front of the incident wave field, and then using it to mathematically propagate the wave field further. This operation is similar to the method of characteristics for partial differential equations, with the added feature of determining the coefficients from the jump in the scattered field at the wave front. The algorithms are also related to the Levinson algorithm of linear least-squares estimation theory; they exploit the Hankel structure in the integral equations, just as the Levinson algorithm exploits the Toeplitz structure in the Yule-Walker equations of linear prediction.

C. Original Research Objectives

The main goal of this project was to develop fast algorithms for solving the multidimensional inverse scattering problem of reconstructing the potential of a Schrodinger equation from various types of scattering data. To this end, several specific tasks are now defined:

1. To mathematically develop the layer stripping idea (heretofore only defined differentially) into a specific numerical algorithm, and to test it with respect to the following items:
   a. Success in reconstructing the scattering potential;
   b. Improvement over (simpler) Born approximation reconstructions;
   c. Robustness of results to small amounts of additive noise in the data;
   d. Robustness of results with respect to variations in discretization grid size;

2. To develop algorithms for generating the scattering data to which the layer stripping algorithm can be applied, i.e., solving the **forward** problem of computing scattering data from a known scattering potential;
3. To develop other iterative algorithms which might be applicable to solving the Marchenko and Gel'fand-Levitan integral equations noted above. This is necessary when the scattering data consists of a far-field scattering amplitude.

D. Additional Research Objectives

In the course of pursuing the research required to achieve the above goals, several other issues arose. In some cases these issues, which were not part of our original proposal, turned out to be more significant than the goals specified above. In other cases serendipitous results were obtained that are of considerable interest in their own right. These included the following:

1. To investigate the *admissibility* of scattering data for inverse scattering problems. We have discovered that this is a crucial issue in applying layer stripping algorithms, and likely a highly significant issue in applying other types of algorithms as well. This work is being developed further in our renewal grant;
2. To investigate how to apply lateral regularization in the layer stripping algorithm. Downward continuation methods for extrapolating wave fields defined on planes through a medium are known to be very ill-conditioned, since the downward continuation problem itself is ill-conditioned. Clearly some sort of regularization is needed to ensure a stable algorithm. How should this be done, and more importantly, what regularization is *implicitly* being applied when the algorithm is discretized, as it must be;
3. To investigate how the wavelet transform could be used to implement layer stripping ideas. The idea here is to use the wavelet transform as an orthonormal basis for representing continuous functions of time and space. We have shown that in the wavelet transform representation wave propagation in a layered one-dimensional medium can be represented by a set of coupled discrete wave systems whose wave speeds are powers of two. This suggests that discrete layer stripping ideas can be applied to continuous

problems in ways other than by direct discretization. Our preliminary results here resulted in an AASERT grant under which this is being investigated further;

4. A major concern of ONR is reconstruction of dielectric media from their reflection response. Since dielectric media are usually absorbing media (waves propagating through them are attenuated due to energy absorption), loss must be included in the forward and inverse scattering problem formulations;
5. To investigate algorithms which employ the above ideas and apply them to closely-related inverse scattering problems, including diffraction tomography, X-ray tomography, and positron-emission tomography.

II. SUMMARY OF RESEARCH ACCOMPLISHMENTS

A. Original Research Objectives

Our accomplishments in answering the original goals can be summarized as follows (the numbering matches that of the goals in Section I):

1. We have successfully developed a layer stripping algorithm for the 2-D inverse scattering problem (task #1), and tested it with respect to the factors listed under task #1. The algorithm works quite well, and successfully reconstructs features of the scattering potential that the Born approximation algorithm is unable to reconstruct. Details are given in Appendix A;
2. We have successfully implemented an invariant imbedding algorithm for computing forward problem data sets for the 2-D inverse scattering problem (task #2). This algorithm is computationally intensive, since it does not use the layer stripping idea (using the same concept to generate forward data and then solve the inverse problem would raise the issue of whether one algorithm was simply running the algorithm backwards). The algorithm is reviewed in Appendix A. The computation time required sparked our interest in iterative algorithms for solving the integral equations arising in the forward problem;
3. We have developed the generalized Landweber iteration into a useful algorithm for solving large systems of equations. We have studied the numerical behavior of this algorithm in detail, discovered some important, new, useful properties, developed a new convergence acceleration procedure, and shown how to control the filtering and convergence properties of this algorithm. Results are in Appendices F-H. Although the specific application considered there is the inverse problem of positron emission tomography, the algorithm should also prove useful in solving the large systems of equations arising from discretized forms of the integral equations of forward and inverse scattering (task #3).

B. Additional Research Objectives

Our accomplishments in answering the additional tasks can be summarized as follows (the numbering matches that of the goals in Section I):

1. The data admissibility issue is discussed in Subsection IIC below.
2. Downward continuation, which all 2-D layer stripping algorithms utilize in one way or another, is *ill-conditioned*, meaning that a small perturbation of the scattering data results in a huge change in the extrapolated wave field at depth. Hence some sort of *regularization* is required. By regularization, we mean that the problem must be altered slightly to remove the ill-conditioning, but the solution to the altered problem must retain the essential characteristics of the solution to the original problem.

   This issue was raised in our paper Appendix A. We now note, in Appendix B, some justifications for the lateral smoothing regularization used in the algorithm of Appendix A. In particular, we note that an explicitly discrete formulation of the 2-D inverse scattering problem leads to the lateral smoothing regularization used by the discretized algorithm of Appendix A, so that the regularization is consistent with the discrete nature of the algorithm.
3. We have applied the wavelet transform, a major hot current research area in signal processing, to the 1-D inverse scattering problem. We have obtained three algorithms for this problem. One is a layer stripping algorithm that operates in the 1-D time-wavelet domain, one is a layer stripping algorithm that operates in the 2-D space-time wavelet domain, and the third is a linear system of equations that comes from the Krein integral equation in the 2-D space-time wavelet domain. Results are in Appendix C;
4. We have developed layer stripping algorithms for 1-D absorbing media and applied these algorithms to problems in reconstruction of absorbing dielectric media from their reflection responses. We have also applied them to reconstruction of lossy transmission lines and to the synthesis of dielectric waveguides. In fact, a complete theory for 1-D forward and inverse scattering problems for discrete layered lossy media has been developed. By a "complete theory" we mean systems of equations that are discrete counterparts to integral equations, and discrete fast algorithms that solve these systems of equations. Results are presented in Appendix E1.

   In addition, we have solved the problem of using plane wave reflection response at two

angles of incidence, rather than both reflection and transmission data (the latter would not be available in remote sensing applications). This leads to a novel semi-iterative use of layer stripping. Numerical examples on reconstructing a glacial ice shelf from radar reflections demonstrate the significance of modelling multiple reflections and losses. Results are presented in Appendix E2.

5. Multiresolution methods for the inverse Radon transform are discussed in Subsection IID below.

C. Data Admissibility

There has been considerable resistance to the idea of layer stripping in general, due to the belief that it is inherently numerically unstable in the presence of noise. This stems from the reputation, dating back to the 1950s, that what was then called "dynamic deconvolution" was numerically unstable in noise, as confirmed by numerical simulations.

However, the Schur algorithm, which is the most basic form of layer stripping, is known to be numerically stable. Furthermore, our basic multidimensional layer stripping algorithm has been shown to be numerically stable in small amounts of additive noise (see Appendix A). This seems to contradict the conventional wisdom noted above.

This contradiction can be resolved by noting that a major concept that has received insufficient attention is the *admissibility* of the scattering data from which the scattering potential is to be reconstructed. A data set is *admissible* if there exists a scattering potential which would generate this (noisy) data set, *in the absence of noise.* An inadmissible data set could not possibly have arisen from *any* scattering potential, and so must be "wrong." A noisy but admissible data set will result in reconstruction of a "noisy" scattering potential, but at least there *is* a potential which can be associated with the noisy data.

The significance of admissibility of data is as follows. First, recall that layer stripping algorithms can be viewed as fast algorithm solutions of integral equations with Toeplitz or Hankel structure. This has been well established in the 1-D case (see Bruckstein, Levy, and Kailath, *SIAM J. Applied Math.* 45, 312-335 (1985)) and extended to the 3-D case in a series of papers by the Principal Investigator. Second, recall that these algorithms

reconstruct the potential recursively, so that they solve a series of subproblems on the way to solving the final problem. This requires not only that the integral equation kernel be positive definite, but that the kernels associated with the subproblems also be positive definite. Finally, it can be shown that these conditions are necessarily true for correct scattering data.

Hence, if these conditions are not fulfilled, some of the subproblems to be solved by the layer stripping algorithm will not be solvable, and the algorithm will fail, by becoming unstable numerically. This is a classic case of "garbage in, garbage out"–if the data set is infeasible, it is *nonsensical*, and the "fault" of the layer stripping algorithm is that it cannot make sense out of nonsense. The algorithm *should* fail, and does. Indeed, *any* algorithm that succeeds on this data set must either: (1) be making some approximation under which the data set becomes admissible (e.g., the Born approximation); or (2) somehow renders the inadmissible data admissible.

This raises the issue of how to make an inadmissible data set admissible, i.e., how should noisy scattering data be altered so that it is admissible (although still noisy)? If this can be done easily, layer stripping algorithms applied to the admissible data will be numerically stable. The reconstructed potential will still be noisy, but at least it can be reconstructed.

We make a start at answering these issues in Appendix B. Specifically, we derive a feasibility condition on 1-D and 2-D free-surface reflection responses from scattering media. We present numerical examples that demonstrate how:

1. The Born approximation is insufficient to reconstruct strongly scattering media, since it neglects multiple scattering;
2. Noise added to the data can make layer stripping algorithms diverge ("blow up");
3. Rendering the noisy data admissible makes the layer stripping algorithm stable. The reconstruction is still noisy, but this is because the data are also still noisy. However, noisy data need not make the algorithm diverge, if the noisy data are still admissible.

D. Multiresolution Methods for Inverting the Radon Transform

We have not totally forsaken the Born approximation under this grant. The 2-D inverse scattering problem with the Born approximation becomes the problem of reconstruction from projections, i.e., inverting the Radon transform. This problem has many applications in medical imaging (where it *is* the problem of x-ray tomography) and nondestructive evaluation (NDE).

This project has used time-frequency methods, especially the wavelet transform, to the following three problems in image reconstruction from projections:

1. We have derived a fast image-domain filter which solves the following constrained inverse Radon transform problem: Given constraints on certain wavelet coefficients of the image, compute from its projections the image which either: (a) requires the smallest perturbation of the projection data to satisfy these constraints; or (b) is the constrained linear least-squares image estimate. The wavelet transform can be used for spatially-varying filtering of an image, suppressing noise locally in smooth regions; we also discuss detection of such regions in a noisy image, which leads to the wavelet coefficient constraints. Numerical results show improvement over filtered images, since the constraints improve the reconstruction in non-constrained areas as well. These results are presented in Appendices J1, J2, and O.
2. We have shown that when the extent of missing angles is small in limited-angle tomography, two of the three sets of detail images in the wavelet transform are unaffected, and low-resolution images can be obtained by interpolation. Using some a priori partial information on edges parallel to the missing angles, we have developed a wavelet-domain algorithm for restoring the image. These results are presented in Appendix M.
3. We have shown that the local tomography problem of reconstructing only a small region of interest (ROI) from a limited set of projections can be solved by sampling the projections at a rate that decreases exponentially with distance from the ROI. This reconstructs the ROI with high resolution, and the remainder of the image at lower resolution. The algorithm is also much faster than conventional filtered backprojection. These results are presented in Appendix N.
4. We have developed a layer-stripping-type algorithm for the causal formulation of the

problem of reconstructing a function from certain values of its spherical means. We have also shown that this problem has an important new application in diffraction tomography. Results are in Appendix D.

5. We have developed a parallel implementation of the 1-D invariant imbedding algorithms. Error analysis of this algorithm shows that the error is the same order of magnitude as the discretization error. However, the resulting algorithm is highly parallelizable. Results are presented in Appendix T.

III. PUBLICATIONS SUPPORTED BY GRANT #N00014-90-J-1897

A. Published Papers

1. A.E. Yagle and P. Raadhakrishnan, "Numerical Performance of Layer Stripping Algorithms for Two-Dimensional Inverse Scattering Problems," Inverse Problems 8(4), 645-665, August 1992. Included as Appendix A.
2. A.E. Yagle, "Inversion of Spherical Means Using Geometric Inversion and Radon Transforms," Inverse Problems 8(6), 949-964, December 1992. Included as Appendix D.
3. T.-S. Pan and A.E. Yagle, "Acceleration and Filtering in the Generalized Landweber Iteration using a Variable Shaping Matrix," IEEE Trans. Medical Imaging 12(2), 278-286, June 1993. Included as Appendix F1.
4. T.-S. Pan and A.E. Yagle, "Acceleration of Landweber-Type Algorithms by Suppression of Projection on the Maximum Singular Vector," IEEE Trans. Medical Imaging 11(4), 479-487, December 1992. Included as Appendix G1.
5. T.-S. Pan and A.E. Yagle, "Numerical Study of Multigrid Implementations of Some Iterative Image Reconstruction Algorithms," IEEE Trans. Medical Imaging 10(4), 572-588, December 1991. Included as Appendix H1.
6. P. Raadhakrishnan, A.E. Yagle, B.V. Rao, and J.E. Dorband, "On Upper Bounds of the Equivalent Oscillator and Notch-Filter Circuits: A Non-Commutative Group Theoretic Approach," IEEE Trans. Circuits and Systems I 39(9), 756-759, Sept. 1992. Included as Appendix I.
7. B. Sahiner and A.E. Yagle, "Image Reconstruction from Projections Under Wavelet Constraints," IEEE Trans. Sig. Proc. 41(12), 3579-3584, December 1993 (special issue on wavelets). Included as Appendix J1.
8. B. Sahiner and A.E. Yagle, "Time-Frequency Distribution Inversion of the Radon Transform," IEEE Trans. Image Proc. 2(4), 539-543, October 1993. Included as Appendix K1.
9. B. Sahiner and A.E. Yagle, "A Fast Algorithm for Backprojection with Linear In-

terpolation," IEEE Trans. Image Proc. 2(4), 547-550, October 1993. Included as Appendix L1.

10. H. Soltanian-Zadeh, J.P. Windham, D.J. Peck, and A.E. Yagle, "A Comparative Analysis of Several Transformations for Enhancement and Segmentation of Magnetic Resonance Image Scene Sequences," IEEE Trans. Medical Imaging 11(3), 302-318, September 1992. Included as Appendix P.
11. H. Soltanian-Zadeh, J.P. Windham and A.E. Yagle, "Optimal Transformation for Correcting Partial Volume Averaging Effects in Magnetic Resonance Imaging," IEEE Trans. Nuclear Science 40(4), 1204-1212, August 1993. Included as Appendix Q1.
12. H. Soltanian-Zadeh, R. Saigal, J.P. Windham, A.E. Yagle, and D.O. Hearshen, "Optimization of MRI Protocols and Pulse Sequence Parameters for Eigenimage Filtering," IEEE Trans. Med. Imag. 13(1), 161-175, March 1994. 1993. Included as Appendix R1.
13. H. Soltanian-Zadeh, J.P. Windham and A.E. Yagle, "A Multidimensional Non-Linear Edge-Preserving Filter for Magnetic Resonance Image Restoration," to appear in IEEE Trans. Image Proc. 4(1), February 1995. Included as Appendix S.
14. J. Frolik and A.E. Yagle, "Reconstruction of Multi-Layered Lossy Dielectrics from Plane Wave Impulse Responses at Two Angles of Incidence," IEEE Trans. Geosci. and Rem. Sensing 33(2), 268-279, March 1995. Included as Appendix E2.
15. B. Sahiner and A.E. Yagle, "Region-of-Interest Tomography using Exponential Radial Sampling," to appear in IEEE Trans. Image Proc. 4(8), August 1995. Included as Appendix N.
16. B. Sahiner and A.E. Yagle, "Reconstruction from Projections under Time-Frequency Constraints," to appear in IEEE Trans. Med. Imag. 14(2), June 1995. Included as Appendix O.
17. P. Raadhakrishnan, J. Dorband, and A.E. Yagle, "An Algorithm for Forward and Inverse Scattering in the Time Domain," to appear in J. Acoust. Soc. Am., April 1995. Included as Appendix T.

B. Conference Papers

1. T.-S. Pan and A.E. Yagle, "Acceleration and Filtering in the Generalized Landweber Iteration using a Variable Shaping Matrix," IEEE 1991 Medical Imaging Conference, Santa Fe, NM, Nov. 5-9, 1991, pp. 2028-2032. Included as Appendix F2.
2. T.-S. Pan and A.E. Yagle, "Acceleration of Landweber-Type Algorithms by Suppression of Projection on the Maximum Singular Vector," IEEE 1991 Medical Imaging Conference, Santa Fe, NM, Nov. 5-9, 1991, pp. 2023-2027. Included as Appendix G2.
3. T.-S. Pan and A.E. Yagle, "Numerical Study of Multigrid Implementations of Some Iterative Image Reconstruction Algorithms," IEEE 1991 Medical Imaging Conference, Santa Fe, NM, Nov. 5-9, 1991, pp. 2033-2037. Included as Appendix H2.
4. B. Sahiner and A.E. Yagle, "On the Use of Wavelets in Inverting the Radon Transform," IEEE 1992 Medical Imaging Conference, Orlando, FL, Oct. 25-31, 1992, pp. 1129-31. Included as Appendix J2.
5. B. Sahiner and A.E. Yagle, "Time-Frequency Distribution Inversion of the Radon Transform," IEEE 1991 Medical Imaging Conference, Santa Fe, NM, Nov. 5-9, 1991, pp. 2043-2047. Included as Appendix K2.
6. B. Sahiner and A.E. Yagle, "A Fast Algorithm for Backprojection," IEEE 1992 Medical Imaging Conference, Orlando, FL, Oct. 25-31, 1992, pp. 1169-71. Included as Appendix L2.
7. B. Sahiner and A.E. Yagle, "Limited Angle Tomography using the Wavelet Transform," IEEE 1993 Medical Imaging Conference, San Francisco, Nov. 4-6, 1993.
8. H. Soltanian-Zadeh, J.P. Windham and A.E. Yagle, "Optimal Transformation for Correcting Partial Volume Averaging Effects in Magnetic Resonance Imaging," IEEE 1992 Medical Imaging Conference, Orlando FL, Oct. 25-31, 1992, pp. 1289-91. Included as Appendix Q2.
9. H. Soltanian-Zadeh, A.E. Yagle, J.P. Windham, and D.O. Hearshen, "Optimization of MRI Protocols and Pulse Sequence Parameters for Eigenimage Filtering," IEEE 1992 Medical Imaging Conference, Orlando, FL, Oct. 25-31, 1992, pp. 1325-27. Included as Appendix R2.

C. Conference Presentations

1. A.E. Yagle, "Multiresolution Algorithms for One-Dimensional Inverse Scattering Problems using the Wavelet Transform," SIAM Conference on Linear Algebra in Signals, Systems, and Control, Seattle, August 16-19, 1993.
2. B. Sahiner and A.E. Yagle, "Constrained Image Reconstruction from Projections using the Wavelet Transform," SIAM Conference on Linear Algebra in Signals, Systems, and Control, Seattle, August 16-19, 1993.
3. H. Soltanian-Zadeh, J.P Windham, D.J. Peck, and A.E. Yagle, "Performance Analysis of Several Transformations for Enhancement and Segmentation of Magnetic Resonance Image Scene Sequences," 33rd Annual Meeting of the American Association of Physicists in Medicine, San Francisco, CA, July 1991.
4. T.-S. Pan and A.E. Yagle, "Numerical Study of Multigrid Implementations of Iterative Image Resonstruction Algorithms," Society of Nuclear Medicine 38th Annual Meeting, Cincinnati OH, June 11-14, 1991.

ALSO: Andrew E. Yagle, "Numerical Performance of Layer Stripping Algorithms for Two-Dimensional Inverse Scattering Problems," presented at Naval Oceanographic and Atmospheric Research Laboratory (NOARL), Stennis Space Center, Bay St. Louis, Mississippi, October 28, 1991.

D. Submitted Papers

1. A.E. Yagle, "On the Feasibility of Impulse Reflection Response Data for the Two-Dimensional Inverse Scattering Problem," submitted to IEEE Trans. Antennas and Propagation. Included as Appendix B.
2. J. Frolik and A.E. Yagle, "Forward and Inverse Scattering for Discrete Layered Lossy and Absorbing Media," submitted to IEEE Trans. Circuits and Systems. Included as Appendix E1.

E. Theses and Final Defense Dates

1. Tin-Su Pan, **The Generalized Landweber Iteration in Positron Emission Tomography**, Ph.D. Thesis, Dept. of EECS, The University of Michigan, Ann Arbor,

MI, June 18, 1991.

2. Hamid Soltanian-Zadeh, **Multidimensional Signal Processing of Magnetic Resonance Imaging Scene Sequences**, January 16, 1992 (Co-Chairman with Dr. Joe Windham, Henry Ford Hospital, Detroit, MI)
3. Berkman Sahiner, **Multiresolution Methods for the Inverse Radon Transform**, Sept. 29, 1993

IV. SUMMARIES OF APPENDICES

Technical details of our research accomplishments are provided in the Appendices. Each of these is a published journal article or conference paper or a submitted journal article. They include details of the problem formulations, the results, and their applications and significance. We now briefly summarize their contents. All of the results presented below are new, unless otherwise indicated.

APPENDIX A

A.E. Yagle and P. Raadhakrishnan, "Numerical Performance of Layer Stripping Algorithms for Two-Dimensional Inverse Scattering Problems," Inverse Problems 8(4), 645-665, August 1992.

This paper summarizes our results to date on the numerical performance of the 2-D layer stripping algorithm with respect to the factors listed in subsection IC-2. It also includes a derivation of the 2-D invariant imbedding algorithm used to produce the forward problem scattering data; this algorithm is based on a previous algorithm (see references). Since the forward problem and inverse problem algorithms operate differently, this constitutes a much better test of the inversion algorithm. A problem with this algorithm is the enormous amount of computation required to generate the forward problem data; our inverse problem algorithm is MUCH faster, even though the inverse problem is mathematically more difficult. The Born approximation to the layer stripping algorithm is also derived and discussed.

The reconstructed potentials agree closely with the original potentials, even for coarse (16×16) discretization grids; the agreeement becomes almost exact for a (64×64) grid. This clearly demonstrates that the layer-stripping concept is numerically viable.

APPENDIX B

A.E. Yagle, "On the Feasibility of Impulse Reflection Response Data for the Two-Dimensional Inverse Scattering Problem," submitted to IEEE Trans. Antennas and Propagation

The contents of this paper are described in some detail in Section IIC.

APPENDIX C

A.E. Yagle, "Multiresolution Algorithms for Solving One-Dimensional Inverse Scattering Problems Using the Wavelet Transform,"

This paper applies the wavelet transform to the 1-D inverse scattering problem. We derive a layer stripping algorithm that uses as input the wavelet transform of the impulse reflection response of the medium. The wavelet transform decouples the layer stripping algorithm into a set of wave systems at differing wave speeds. Any of these multiple-resolution systems could be used to reconstruct the medium; this allows some flexibility on how the data are used.

We also derive both a layer stripping algorithm, and a linear system of equations, in the 2-D (time and space) wavelet transform domain, from the layer stripping algorithm and Krein integral equation, respectively. These results show how data at one resolution affects the reconstruction at another resolution. They are interpreted in terms of fast algorithms for the slanted Toeplitz structured linear system of equations. The Born approximation is also derived and discussed.

This paper formed the basis of a successful AASERT proposal to ONR.

APPENDIX D

A.E. Yagle, "Inversion of Spherical Means Using Geometric Inversion and Radon Transforms," Inverse Problems 8(6), 949-964, December 1992.

This paper analyzes the problem of reconstructing a function from its spherical means passing through the origin. A new application of this problem to diffraction tomography is noted: We show that given probing by impulsive plane waves at all angles of incidence, only a single receiving sensor is necessary, not an array of sensors.

Two versions of the problem are defined. A layer-stripping-type algorithm is derived for one version (we use the term invariant imbedding in the paper, since it is more familiar to readers, but it is really layer stripping). The two versions are shown to be equivalent to the usual and exterior inverse Radon transforms, respectively, using geometric inversion (reflection about a circle). A simple numerical example is also included.

APPENDIX E1

J. Frolik and A.E. Yagle, "Forward and Inverse Scattering for Discrete Layered Lossy and Absorbing Media," submitted to IEEE Trans. Circuits and Systems

A complete theory for the 1-D forward and inverse scattering problems for discrete layered lossy media is presented. By a "complete theory," we mean systems of equations that are discrete counterparts to integral equations, and discrete fast algorithms that solve these systems of equations. Applications to discrete lossy transmission lines, and to electromagnetic wave propagation in absorbing dielectrics, are made, and numerical examples presented.

APPENDIX E2

J. Frolik and A.E. Yagle, "Reconstruction of Multi-Layered Lossy Dielectrics from Plane Wave Impulse Responses at Two Angles of Incidence," IEEE Trans. Geosci. and Rem. Sensing 33(2), 268-279, March 1995.

The problem posed in Appendix E1 is solved using plane wave reflection response at two angles of incidence, rather than reflection and transmission data (the latter would not be available in remote sensing applications). This includes a novel seni-iterative use of layer stripping. Numerical examples on reconstructing a glacial ice shelf from radar reflections demonstrate the significance of modelling multiple reflections and losses.

APPENDIX F1

T.-S. Pan and A.E. Yagle, "Acceleration and Filtering in the Generalized Landweber Iteration using a Variable Shaping Matrix," IEEE Trans. Medical Imaging 12(2), 278-286, June 1993.

This paper discusses the generalized Landweber iteration for solving large linear systems of equations. We show how the convergence and filtering behavior of this algorithm can be tightly controlled, in contrast to most iterative algorithms whose behavior cannot be controlled and that simply go where they may. This paper is a good introduction to the algorithm and how to design it to obtain desired behavior.

Although the specific application investigated here is positron emission tomography, the results could also be applied to discretized integral equations, such as the generalized Gel'fand-Levitan or Marchenko integral equations. Although the matrix kernels are no longer sparse, a projection or backprojection (multiplication by the kernel or its transpose) can be implemented quickly using FFT-based convolution methods; number-theoretic transforms would require even fewer multiplications.

APPENDIX F2

T.-S. Pan and A.E. Yagle, "Acceleration and Filtering in the Generalized Landweber Iteration using a Variable Shaping Matrix," IEEE 1991 Medical Imaging Conference, Santa Fe, NM, Nov. 5-9, 1991, pp. 2028-2032.

This is the conference paper version of Appendix F1.

APPENDIX G1

T.-S. Pan and A.E. Yagle, "Acceleration of Landweber-Type Algorithms by Suppression of Projection on the Maximum Singular Vector," IEEE Trans. Medical Imaging 11(4), 479-487, December 1992.

This paper presents a simple procedure to significantly accelerate the convergence of the generalized Landweber iteration.

APPENDIX G2

T.-S. Pan and A.E. Yagle, "Acceleration of Landweber-Type Algorithms by Suppression of Projection on the Maximum Singular Vector," IEEE 1991 Medical Imaging Conference, Santa Fe, NM, Nov. 5-9, 1991, pp. 2023-2027.

This is the conference paper version of Appendix G1.

APPENDIX H1

T.-S. Pan and A.E. Yagle, "Numerical Study of Multigrid Implementations of Some Iterative Image Reconstruction Algorithms," IEEE Trans. Medical Imaging 10(4), 572-588, December 1991.

This paper investigates the use of several iterative algorithms, including the gener-

alized Landweber iteration, in multigrid image reconstruction. The image is first reconstructed quickly on a coarse grid. This coarse image is then used as the initialization for reconstruction of the image on a fine grid. Many numerical examples are used to illustrate the performance of various algorithms.

APPENDIX H2

T.-S. Pan and A.E. Yagle, "Numerical Study of Multigrid Implementations of Some Iterative Image Reconstruction Algorithms," IEEE 1991 Medical Imaging Conference, Santa Fe, NM, Nov. 5-9, 1991, pp. 2033-2037.

This is the conference paper version of Appendix H1.

APPENDIX I

P. Raadhakrishnan, A.E. Yagle, B.V. Rao, and J.E. Dorband, "On Upper Bounds of the Equivalent Oscillator and Notch-Filter Circuits: A Non-Commutative Group Theoretic Approach," IEEE Trans. Circuits and Systems I 39(9), 756-759, Sept. 1992.

This paper is a minor work that uses group theory to aid in designing oscillator circuits. The application is quite unusual and novel.

APPENDIX J1

B. Sahiner and A.E. Yagle, "Image Reconstruction from Projections Under Wavelet Constraints," IEEE Trans. Sig. Proc. 41(12), 3579-3584 December 1993 (special issue on wavelets).

This paper considers the problem of image reconstruction from projections, given constraints not on the image, but on certain wavelet coefficients of the image. The idea is that low-resolution regions of the image can be locally low-pass filtered by setting high-resolution wavelet coefficients to zero. These are then used as constraints on the image reconstruction process, so that other areas of the reconstructed image are improved as well. The constraints are implemented as a simple filter directly on the image.

APPENDIX J2

B. Sahiner and A.E. Yagle, "On the Use of Wavelets in Inverting the Radon Transform," IEEE 1992 Medical Imaging Conference, Orlando, FL, Oct. 25-31, 1992, pp. 1129-31.

This is the conference paper version of Appendix J1.

APPENDIX K1

B. Sahiner and A.E. Yagle, "Time-Frequency Distribution Inversion of the Radon Transform," IEEE Trans. Image Proc. 2(4), 539-543, October 1993.

This paper performs a time-frequency analysis of the projection data in the inverse Radon transform problem. Regions in time-frequency space in which the distribution strength is below a threshold are assumed to be due to noise, and are set to zero. This has the effect of filtering noise out of time-frequency regions in which the signal strength is small, and leaving the noise in where the signal strength is large. The resulting time-frequency distribution is then projected to find the nearest feasible signal solution, which is then backprojected. This reduces noise in the reconstructed image while maintaining sharpness of image features.

APPENDIX K2

B. Sahiner and A.E. Yagle, "Time-Frequency Distribution Inversion of the Radon Transform," IEEE 1991 Medical Imaging Conference, Santa Fe, NM, Nov. 5-9, 1991, pp. 2043-2047.

This is the conference paper version of Appendix K1.

APPENDIX L1

B. Sahiner and A.E. Yagle, "A Fast Algorithm for Backprojection with Linear Interpolation," IEEE Trans. Image Proc. 2(4), 547-550, October 1993.

This paper derives a simple fast algorithm for backprojection in the inverse Radon transform. Interpolating and backprojecting four views at once saves half the multiplications.

APPENDIX L2

B. Sahiner and A.E. Yagle, "A Fast Algorithm for Backprojection," IEEE 1992 Medical Imaging Conference, Orlando, FL, Oct. 25-31, 1992, pp. 1169-71.

This is the conference paper version of Appendix L1.

APPENDIX M

B. Sahiner and A.E. Yagle, "Limited Angle Tomography Using the Wavelet Transform," revision submitted to IEEE Trans. Image Proc.

This paper shows that when the extent of missing angles is small in limited-angle tomography, two of the three sets of detail images in the wavelet transform are unaffected, and low-resolution images can be obtained by interpolation. Using some a priori partial information on edges parallel to the missing angles, we have developed a wavelet-domain algorithm for restoring the image.

APPENDIX N

B. Sahiner and A.E. Yagle, "Region-of-Interest Tomography using Exponential Radial Sampling," to appear in IEEE Trans. Image Proc. 4(9), August 1995.

This paper shows that the local tomography problem of reconstructing only a small region of interest (ROI) from a limited set of projections can be solved by sampling the projections at a rate that decreases exponentially with distance from the ROI. This reconstructs the ROI with high resolution, and the remainder of the image at lower resolution. The algorithm is also much faster than conventional filtered backprojection.

APPENDIX O

B. Sahiner and A.E. Yagle, "Reconstruction from Projections under Time-Frequency Constraints," to appear in IEEE Trans. Med. Imag. 14(2), June 1995.

This paper derives a fast image-domain filter which solves the following constrained inverse Radon transform problem: Given constraints on certain wavelet coefficients of the image, compute from its projections the image which either: (a) requires the smallest perturbation of the projection data to satisfy these constraints; or (b) is the constrained

linear least-squares image estimate. The wavelet transform can be used for spatially-varying filtering of an image, suppressing noise locally in smooth regions; we also discuss detection of such regions in a noisy image, which leads to the wavelet coefficient constraints. Numerical results show improvement over filtered images, since the constraints improve the reconstruction in non-constrained areas as well.

APPENDIX P

H. Soltanian-Zadeh, J.P. Windham, D.J. Peck, and A.E. Yagle, "A Comparative Analysis of Several Transformations for Enhancement and Segmentation of Magnetic Resonance Image Scene Sequences," IEEE Trans. Medical Imaging 11(3), 302-318, September 1992.

Its title describes this paper very well.

APPENDIX Q1

H. Soltanian-Zadeh, J.P. Windham and A.E. Yagle, "Optimal Transformation for Correcting Partial Volume Averaging Effects in Magnetic Resonance Imaging," IEEE Trans. Nuclear Science 40(4), 1204-1212, August 1993.

Again, its title describes this paper very well.

APPENDIX Q2

H. Soltanian-Zadeh, J.P. Windham and A.E. Yagle, "Optimal Transformation for Correcting Partial Volume Averaging Effects in Magnetic Resonance Imaging," IEEE 1992 Medical Imaging Conference, Orlando FL, Oct. 25-31, 1992, pp. 1289-91.

This is the conference paper version of Appendix Q1.

APPENDIX R1

H. Soltanian-Zadeh, R. Saigal, J.P. Windham, A.E. Yagle, and D.O. Hearshen, "Optimization of MRI Protocols and Pulse Sequence Parameters for Eigenimage Filtering," IEEE Trans. Medical Imaging 13(1), 161-175, March 1994.

This paper proposes a procedure for optimizing the acquisition of MRI scene se-

quences, if eigenimage filtering (see Appendix P) is then used to process the MRI scene sequence.

APPENDIX R2

H. Soltanian-Zadeh, A.E. Yagle, J.P. Windham, and D.O. Hearshen, "Optimization of MRI Protocols and Pulse Sequence Parameters for Eigenimage Filtering," IEEE 1992 Medical Imaging Conference, Orlando, FL, Oct. 25-31, 1992, pp. 1325-27.

This is the conference paper version of Appendix R1.

APPENDIX S

H. Soltanian-Zadeh, J.P. Windham and A.E. Yagle, "A Multidimensional Non-Linear Edge-Preserving Filter for Magnetic Resonance Image Restoration," IEEE Trans. Image Proc. 4(2), February 1995.

Although the edge-preserving filter with locally-varying properties was designed specifically for MRI, it may have applications elsewhere.

APPENDIX T

P. Raadhakrishnan, J. Dorband, and A.E. Yagle, "An Algorithm for Forward and Inverse Scattering in the Time Domain," to appear in J. Acoust. Soc. Am., April 1995.

As noted above, the invariant imbedding algorithm used to generate the forward problem data is computationally intensive. This is because invariant imbedding, although similar to layer stripping in approach, is quite different, in that it does not take advantage of time causality. Hence it solves many more problems than are actually needed. Layer stripping avoids this and is more efficient.

This paper is a first attempt at parallelizing invariant-imbedding-based algorithms for both the forward and the inverse problems, and thus reducing the computation required. Only the 1-D problem is considered here. The new algorithm is parallelizable and gives results identical to the invariant imbedding algorithm of Corones et al. An simple error analysis of the effects of computational noise on the algorithm is also supplied.

APPENDIX A

A.E. Yagle and P. Raadhakrishnan, "Numerical Performance of Layer Stripping Algorithms for Two-Dimensional Inverse Scattering Problems," Inverse Problems 8(4), 645-665, August 1992.

This paper summarizes our results to date on the numerical performance of the 2-D layer stripping algorithm with respect to the factors listed in subsection IC-2. It also includes a derivation of the 2-D invariant imbedding algorithm used to produce the forward problem scattering data; this algorithm is based on a previous algorithm (see references). Since the forward problem and inverse problem algorithms operate differently, this constitutes a much better test of the inversion algorithm. A problem with this algorithm is the enormous amount of computation required to generate the forward problem data; our inverse problem algorithm is MUCH faster, even though the inverse problem is mathematically more difficult. The Born approximation to the layer stripping algorithm is also derived and discussed.

The reconstructed potentials agree closely with the original potentials, even for coarse (16×16) discretization grids; the agreeement becomes almost exact for a (64×64) grid. This clearly demonstrates that the layer-stripping concept is numerically viable.

Inverse Problems 8 (1992) 645–665. Printed in the UK

Numerical performance of layer stripping algorithms for two-dimensional inverse scattering problems

Andrew E Yagle and Poovendran Raadhakrishnan

Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109-2122, USA

Received 29 August 1991

Abstract. Numerical results of implementing a two-dimensional layer stripping algorithm to solve the two-dimensional Schrödinger equation inverse potential problem are presented and discussed. This is the first exact (all multiple scattering and diffraction effects are included) numerical solution of a multi-dimensional Schrödinger equation inverse potential problem, excluding optimization-based approaches. The results are as follows: (1) the layer stripping algorithm successfully reconstructed the potential from scattering data measured on a plane (as it would be in many applications); (2) the algorithm avoids multiple scattering errors present in Born approximation reconstructions; and (3) the algorithm is insensitive to small amounts of noise in the scattering data. Simplifications of layer stripping and invariant imbedding algorithms under the Born approximation are also discussed.

1. Introduction

The inverse scattering problem for the Schrödinger equation in two dimensions with a time-independent, local, non-circularly symmetric potential has many applications. Two of these applications are as follows: (1) reconstruction of a three-dimensional (3D) acoustic medium with density and wave speed varying in two dimensions (2D), from surface measurements of the steady-state medium displacement response to a harmonic line source [1]; and (2) reconstruction of a 3D electrical medium with resistivity varying in 2D from surface measurements of the potential resulting from a line DC current source [2]. Both of these applications are quickly reviewed below in section 2.1.

Two major approaches for obtaining exact solutions of the 2D Schrödinger equation inverse potential problem have been proposed. The first is the 2D version of the Gel'fand–Levitan and Marchenko integral equation methods [3]. The other is the 2D version of the layer stripping differential methods [4]. Here 'exact' means that all diffraction and multiple scattering effects are included in the mathematical solution; errors in the solution will arise solely due to purely numerical effects such as discretization and roundoff. Hence all methods based on the Born (single-scattering) approximation are excluded here, since such methods, and their modifications, do not take into account *all* multiple scattering effects. In section 2.4 we discuss how the Born approximation applies to the algorithm of [4]. No numerical implementation of the methods of either [3] or [4] has previously been reported.

0266-5611/92/040645+21$04.50 © 1992 IOP Publishing Ltd

This paper presents the results of the first numerical implementation of the 2D version of the layer-stripping algorithm of [4]. It is thus the first *exact* (as defined above) numerical solution of a multi-dimensional Schrödinger equation inverse potential problem. Note that optimization-based approaches minimize (or maximize) some criterion; thus they are not in the spirit of the approach considered here. Although only reconstruction of the Schrödinger scattering potential is considered here, direct application to specific inverse scattering problems, as in [1] and [2], would be possible.

This paper is organized as follows. In section 2 the 2D Schrödinger equation inverse potential problem is formulated, two applications are noted, the layer stripping algorithm of [4] is reviewed, details of its numerical implementation are discussed, and its simplification under the Born approximation is discussed. In section 3 the invariant imbedding algorithm of [5] used to generate the scattering data is reviewed, and details of its numerical implementation are discussed. We also discuss its simplification under the Born approximation, and show analytically that the Born-simplified layer stripping algorithm successfully inverts the Born-simplified invariant imbedding algorithm scattering data. Although the latter result is new, it is intended primarily to give some feel for the algorithms of [4] and [5].

Section 4 summarizes the numerical results, and presents some illustrative examples. Issues illustrated include: (1) errors in reconstructed potentials using the Born approximation, which are eliminated using the 'exact' layer stripping algorithm; (2) effects on reconstructed potentials of various amounts of noise in the data; (3) effects on reconstructed potentials of regularization of transverse derivatives in the layer stripping algorithm; and (4) effects of choosing various discretization lengths in the layer stripping algorithm. Section 5 concludes with a summary.

2. Two-dimensional layer stripping algorithm

2.1. Problem formulation and applications

The 2D inverse scattering problem considered in this paper is as follows. The problem is defined in 2D (x, z) space, where x is lateral position and z is depth, increasing downward from the surface $z = 0$. The wavefield $\hat{p}(x, z, k)$ satisfies the 2D Schrödinger equation

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial z^2} + k^2 - V(x, z)\right)\hat{p}(x, z, k) = 0 \tag{2.1}$$

where the potential $V(x, z)$ is real-valued, smooth, and has support in z in the interval $0 < z < L$. It is also assumed that $V(x, z)$ does not induce bound states; a sufficient condition for this is for $V(x, z)$ to be non-negative.

The medium is probed by an impulsive plane wave e^{-ikz}, which passes through the surface $z = 0$ at time $t = 0$ and induces scattering by $V(x, z)$ for $t > 0$. The scattering data consists of measurements of the wavefield $\hat{p}(x, z^*, k)$ and its gradient $\partial\hat{p}(x, z^*, k)/\partial z$ for some z^* in the homogeneous half-space $z \geqslant 0$. For convenience, we assume measurements are taken at the surface $z^* = 0$, as they would be in the applications to follow. The inverse scattering experiment is illustrated in figure 1.

We now quickly review two applications of this problem. First, consider the problem of reconstructing a 3D inhomogeneous acoustic medium whose density $\rho(x, z)$

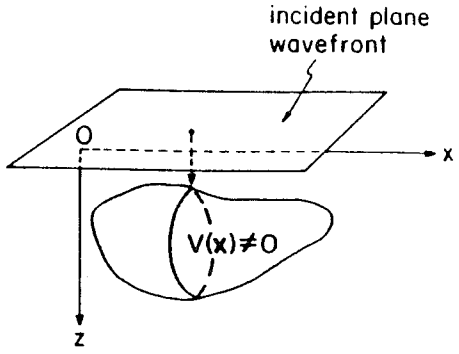


Figure 1. The 2D inverse scattering problem.

and wave speed $c(x, z)$ are smooth functions of depth z and lateral position x. The medium is bounded by a free (pressure-release) surface $z = 0$. The density ρ_0 and wave speed c_0 for $z < 0$ and $z \to \infty$ are known. The medium is probed with cylindrical harmonic waves, at two frequencies ω_1 and ω_2, from a harmonic line source extending along the x-axis, and the sinusoidal steady-state vertical acceleration $\hat{a}(x, y, z = 0; \omega_i)$ of the medium at the free surface $z = 0$ is measured. The goal is to reconstruct $\rho(x, z)$ and $c(x, z)$ from the measurements $a(x, y, z = 0; \omega_i)$, $i = 1, 2$.

This problem can be formulated as a 2D Schrodinger equation inverse potential problem by Fourier transforming the basic acoustic equations with respect to time and the other lateral variable y. Details are given in both [1] and [4]. Here we merely note that in the Schrödinger equation (2.1) the wavefield $p(x, z, k)$ is pressure divided by $\rho(x, z)^{1/2}$, the wavenumber $k^2 = \omega_i^2/c_0^2 - k_y^2$, and the potential $V(x, z; \omega_i)$ is

$$V(x, z; \omega_i) = \left(\frac{\omega_i^2}{c_0^2}\right)\left(1 - \frac{c_0^2}{c(x, z)^2}\right) + \rho(x, z)^{1/2}\nabla^2(\rho(x, z)^{-1/2}). \tag{2.2}$$

It is clear that performing this experiment for two different frequencies ω_i, $i = 1, 2$ will allow $\rho(x, z)$ and $c(x, z)$ to be computed from (2.2). The wavefield is zero at the free surface $z = 0$; its gradient is the medium acceleration $\rho(x, 0)^{1/2}\hat{a}(x, y, z = 0; \omega_i)$, $i = 1, 2$.

The second application is the inverse resistivity problem of reconstructing a 3D inhomogeneous electrical medium whose resistivity $\rho(x, z)$ is a smooth function of x and z over a bounded region. The medium is probed with current from a line DC current source extending along the x-axis, and the electrical potential $v(x, y, z = 0)$ induced on the surface $z = 0$, assumed to be a perfect insulator, is measured. The goal is to reconstruct the resistivity $\rho(x, z)$ from the measurements of electrical potential $v(x, y, z = 0)$. Note that for both applications, the response to a *line* source may be found by superposition of the responses due to *point* sources along the x-axis.

This problem can be formulated as a 2D Schrodinger equation inverse potential problem by Fourier transforming Ohm's and Kirchoff's current laws with respect to the other lateral variable y. Details are given in [5]. Here we merely note that in the Schrödinger equation (2.1) the wavefield $p(x, z, k)$ is now the inverse Laplace

transform of the Fourier transform of electrical potential divided by $\rho(x,z)^{1/2}$, and the scattering potential $V(x,z) = \rho(x,z)^{1/2}\nabla^2(\rho(x,z)^{-1/2})$.

2.2 *The 2D layer stripping algorithm*

The layer stripping algorithm for solving the 2D Schrödinger equation inverse potential problem is derived as follows [4]. Taking the inverse Fourier transform of (2.1) with respect to k yields

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial z^2} - \frac{\partial^2}{\partial t^2} - V(x,z)\right)\check{p}(x,z,t) = 0 \tag{2.3}$$

where

$$\check{p}(x,z,t) = \frac{1}{2\pi}\int_{-\infty}^{\infty}\hat{p}(x,z,k)e^{ikt}\,dk. \tag{2.4}$$

Equation (2.3) can be written as the coupled system

$$\left(\frac{\partial}{\partial z} + \frac{\partial}{\partial t}\right)\check{p}(x,z,t) = \check{q}(x,z,t) \tag{2.5a}$$

$$\left(\frac{\partial}{\partial z} - \frac{\partial}{\partial t}\right)\check{q}(x,z,t) = \left(V(x,z) - \frac{\partial^2}{\partial x^2}\right)\check{p}(x,z,t) \tag{2.5b}$$

From causality and the form of (2.5a), $\check{p}(x,z,t)$ and $\check{q}(x,z,t)$ have the forms

$$\check{p}(x,z,t) = \delta(t-z) + \tilde{p}(x,z,t)1(t-z) \tag{2.6a}$$

$$\check{q}(x,z,t) = \tilde{q}(x,z,t)1(t-z) \tag{2.6b}$$

where $\tilde{p}$ and $\tilde{q}$ are the smooth parts of $\check{p}$ and $\check{q}$, respectively, and $1(\ )$ is the unit step or Heaviside function.

Inserting (2.6) in (2.5) and equating coefficients of $\delta(t-z)$ (propagation of singularities argument) yields

$$\left(\frac{\partial}{\partial z} + \frac{\partial}{\partial t}\right)\tilde{p}(x,z,t) = \tilde{q}(x,z,t) \tag{2.7a}$$

$$\left(\frac{\partial}{\partial z} - \frac{\partial}{\partial t}\right)\tilde{q}(x,z,t) = \left(V(x,z) - \frac{\partial^2}{\partial x^2}\right)\tilde{p}(x,z,t) \tag{2.7b}$$

$$V(x,z) = -2\tilde{q}(x,z,t=z^+). \tag{2.7c}$$

Equations (2.7) constitute the *basic 2D layer stripping equations*: starting with measured $\tilde{p}(x,0,t)$ and $\tilde{q}(x,0,t)$ (the gradient of the wavefield is required for the latter), propagate (2.7) recursively in increasing depth z, reconstructing $V(x,z)$ as the algorithm proceeds. The coupled equations (2.7*a*) and (2.7*b*) include *all* multiple scattering and diffraction effects, since they are equivalent to the Schrödinger equation (2.1) in the time domain. The potential may be reconstructed using (2.7*c*) since

(2.7) is implemented at the wave front $t = z$; by time causality there has been no time for multiple scattering to occur yet.

Some advantages of using layer stripping algorithms are as follows.

(i) Only backscattered data from one direction of probing is required. Integral equation methods [3] require the scattering amplitude, which is the far-field response in *all* directions to an incident impulsive plane wave in *each* possible direction. In the applications noted above, this is unrealistic; it also runs the risk of inconsistent data.

(ii) The amount of computation required is much less than the amount required to solve the integral equations of [3]. The layer stripping algorithm can be viewed as a fast algorithm solution of these integral equations which exploits the Hankel structure in the kernel of the generalized Marchenko integral equation of [3].

(iii) *All* multiple scattering and diffraction effects are included, unlike methods such as distorted-wave Born approximation which only account for some of these effects.

Two disadvantages of layer stripping algorithms are as follows.

(i) It is not clear how to incorporate the effects of bound states (roughly, square-integrable solutions to the Schrödinger equation with negative energy); unlike the approach of [3].

(ii) The lateral derivative $\partial^2/\partial x^2$ in (2.7*b*) can be expected to induce numerical instability.

2.3. *Numerical implementation of the 2D layer stripping algorithm*

The second disadvantage can be removed as follows. Take the Fourier transform of (2.7) with respect to x. The result is

$$\left(\frac{\partial}{\partial z} + \frac{\partial}{\partial t}\right)p(z,t,k_x) = q(z,t,k_x) \tag{2.8a}$$

$$\left(\frac{\partial}{\partial z} - \frac{\partial}{\partial t}\right)q(z,t,k_x) = k_x^2 p(z,t,k_x) + \hat{V}(z,k_x) * p(z,t,k_x) \tag{2.8b}$$

$$\hat{V}(z,k_x) = -2q(z,t=z^+,k_x) \tag{2.8c}$$

where $*$ denotes convolution in k_x

$$p(z,t,k_x) = \int_{-\infty}^{\infty}\tilde{p}(x,z,t)e^{-ik_x x}\,dx \tag{2.9}$$

and $q(z,t,k_x)$ and $\hat{V}(z,k_x)$ are defined similarly.

The multiplication by k_x^2 in (2.8*b*) will induce numerical instability. This may be avoided by replacing the multiplication by k_x^2 in (2.8*b*) with multiplication by the clipped filter

$$H(k_x) = \begin{cases} k_x^2 & \text{if } |k_x| < K \\ 0 & \text{otherwise} \end{cases} \tag{2.10}$$

for some cutoff wavenumber K. This is reminiscent of the clipped filter used in the filtered back-projection procedure for inverting the Radon transform. In practice, the

discontinuities in (2.10) at $|k_x| = K$ would be replaced by a smooth window to zero; a Hanning (raised cosine) window was used in the numerical simulations presented later.

We now discretize depth $z = n\Delta$ and time $t = j\Delta$ to integer multiples of some discretization length Δ. Since the wave speed in (2.1) is unity, depth and time have the same Δ. Lateral position x would also use the same Δ; but wavenumber $k_x = k\Delta_k$ must use for Δ_k half the reciprocal of the total lateral extent of interest; e.g. if the potential has finite support $-L_x/2 < x < L_x/2$ in x, L_x would be the lateral extent of interest. Note that Δ and Δ_k have reciprocal units.

Using forward difference approximations to the partial derivatives then yields

$$p((n+1)\Delta,(j+1)\Delta,k\Delta_k) = p(n\Delta,j\Delta,k\Delta_k) + q(n\Delta,j\Delta,k\Delta_k)\Delta \tag{2.11a}$$

$$q((n+1)\Delta,(j-1)\Delta,k\Delta_k) = q(n\Delta,j\Delta,k\Delta_k) + H(k\Delta_k)p(n\Delta,j\Delta,k\Delta_k)\Delta + \sum_{m=-\infty}^{\infty} \hat{V}((k-m)\Delta_k)p(n\Delta,j\Delta,m\Delta_k)\Delta\Delta_k \tag{2.11b}$$

$$V((n+1)\Delta,k\Delta_k) = -2q((n+1)\Delta,(n+1)\Delta,k\Delta_k). \tag{2.11c}$$

Equations (2.11) constitute the numerical implementation of the 2D layer stripping algorithm. The update patterns are illustrated in figure 2; note that by time causality $p(z,t,k_x)$ and $q(z,t,k_x)$ are zero for $t < z$.

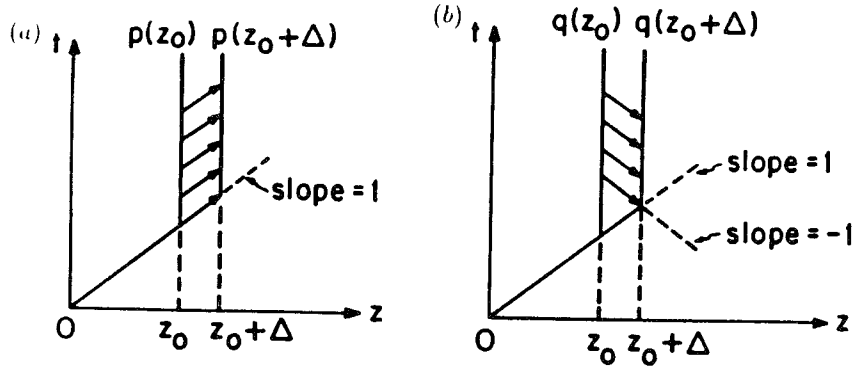


Figure 2. Update patterns for (*a*) $p(z,t,k_x)$ and (*b*) $q(z,t,k_x)$.

Cheney [6] has shown that the modification (2.10) stabilizes the layer stripping algorithm (2.11), in the following sense. Define the norm

$$\|f(k_x,t)\| = \sup_{t>0} \int_{-\infty}^{\infty} |f(k_x,t)|\,dk_x. \tag{2.12}$$

Input two different sets of bounded initial data $p_i(k_x,0,t), q_i(k_x,0,t), i = 1,2$ into the discretized algorithm (2.11), resulting in two different reconstructed potentials

$\hat{V}_i(k_x,z)$, $i = 1,2$. Let $\|p_i(k_x,0,t)\| < K'$ and $\|q_i(k_x,0,t)\| < K'$ for some K'. Then for $z = n\Delta$ we have

$$\sup_{k_x}[\hat{V}_1(k_x,z) - \hat{V}_2(k_x,z)] \leqslant K_1(z)\|p_1(k_x,0,t) - p_2(k_x,0,t)\| + K_2(z)\|q_1(k_x,0,t) - q_2(k_x,0,t)\| \tag{2.13}$$

where $K_1(z)$ and $K_2(z)$ are polynomials in n, Δ, K, and K'.

The discretized system (2.11) can be implemented as is. However, its spectral properties are worth examining. It might seem as though we can regard the discretized functions $p(n\Delta,j\Delta,k\Delta)$, etc, as merely sampled versions of the continuous functions $p(z,t,k_x)$, etc, provided the latter are bandlimited and sampling is performed above the Nyquist rate. However, the nonlinear product in (2.7*b*) becomes the convolution in k_x in (2.8*b*) and (2.11*b*); the wavenumbers become mixed. Indeed, even if the inverse potential problem is regularized by assuming that $V(z,k_x)$ is bandlimited in z and zero for $|k_x| > K$ for some K, it is clear that $p(z,t,k_x)$, etc. will *not* have similar properties. Imposing a bandlimited condition at each recursion will lead to errors, since the missing high wavenumbers will cause errors at low wavenumbers due to the wavenumber mixing. This leads to the question of what the discretized $p(n\Delta,j\Delta,k\Delta_k)$, etc mean, and how the convolutions in k_x should be performed. It should be noted that similar questions arise in integral equation methods.

One possible interpretation is to perform a periodic extension in k of all quantities in (2.11). The period in k should be $1/\Delta_k$; K in (2.10) should then be half this. It is clear by induction that if all quantities at depth $n\Delta$ are periodic in k, then all quantities at depth $(n+1)\Delta$ will also be periodic in k. This has two advantages: (1) the infinite linear convolution becomes a finite cyclic convolution; and (2) the discrete Fourier transform may be used to perform all Fourier transforms. Since periodicity in one Fourier domain is equivalent to discreteness in the other Fourier domain, the problem has effectively been discretized laterally as well as vertically: the quantites propagated in (2.11) are not samples of a bandlimited function, but actual discrete values. As $\Delta_k \to 0$, the situation approaches the continuous problem.

2.4. *Born approximation to the layer stripping algorithm*

It is worth noting how the Born approximation applies to the layer stripping equations (2.7). The Born approximation is a linearization of the inverse potential problem; the idea is to render the potential to be linearly related to the scattering data. This has been discussed in detail elsewhere; here we merely scale the potential by a small parameter ϵ, expand $\tilde{p}(x,z,t)$, etc. in a Taylor series in ϵ, and discard all terms of order ϵ^2 or smaller. The result is elimination of the product in (2.7*b*); since this is the one nonlinearity in (2.7) its elimination is not surprising. Combining the modified (2.7*a*) and (2.7*b*) and keeping (2.7*c*) results in

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial z^2} - \frac{\partial^2}{\partial t^2}\right)\tilde{q}(x,z,t) = 0 \tag{2.14a}$$

$$V(x,z) = -2\tilde{q}(x,z,t=z^+). \tag{2.14b}$$

We recognize (2.14a) as the migration operator relating the wavefield at the surface $z = 0$ to the wavefield on the plane parallel to the surface at depth z, and (2.14b) as the imaging operator (gradient) applied to the migrated wavefield. Taking two Fourier transforms of (2.14a) with respect to t and x and using (2.4) and (2.9) yields

$$\left(\frac{\partial^2}{\partial z^2} + (k^2 - k_x^2)\right)\bar{q}(k_x, z, k) = 0 \qquad \bar{q}(k_x, z, k) = \mathcal{F}_{t\to k}\mathcal{F}_{x\to k_x}\{\tilde{q}(x, z, t)\} \tag{2.15}$$

a differential equation which has the solution

$$\bar{q}(k_x, z, k) = \bar{q}(k_x, 0, k)e^{-i\sqrt{k^2-k_x^2}z}. \tag{2.16}$$

The operation of the Born approximation to the layer stripping equations is now clear: (1) migrate the wavefield from the surface to depth z; and (2) image the wavefield at depth z to obtain the scattering potential. Note that imaging the potential requires taking the gradient of the wavefield; this is why q, not p, is used. Note also that multiple scattering, which is inherently nonlinear, is neglected in (2.14) and (2.16). The coupling induced by the product term in (2.7) accounts precisely for all multiple scattering. More details about the Born approximation and its relation to layer stripping and integral equation methods is available in [4] and [7].

From the Schrödinger equation (2.1), it is apparent that for large wavenumbers k the potential $V(x, z)$ will be relatively small, and that multiple scattering will be less significant. Indeed, in the limit $k \to \infty$ the Born approximation becomes exact, in that multiple scattering effects become negligible. However, inversion based solely on asymptotically large k is clearly unstable; 'exact' inverse scattering methods use low-wavenumber data as well as high-wavenumber data to stabilize the reconstruction. Also, it is clear that multiple scattering is more significant for small k ($V(x, z)$ is relatively large), so lack of high-wavenumber data makes the use of 'exact' methods even more imperative.

3. Forward problem algorithm

3.1. *Invariant imbedding algorithm*

The invariant imbedding algorithm of [5] was used to generate the scattering data, to be input into the layer stripping algorithm. We briefly review this algorithm here, following the notation of [5] for convenience. Let k be wavenumber, as in the Schrödinger equation (2.1), $q = k_x$ (lateral wavenumber), $k(q) = \sqrt{k^2 - q^2}$ (vertical wavenumber, as in (2.16)), and p be lateral wavenumber of the incident plane wave (ultimately we are interested in $p = 0$). Then further define $h(z, q) = \tilde{V}(z, k_x)$ (scattering potential) and $u(z, q) = \bar{p}(k_x, z, k)$ (wavefield; see (2.15)). A slight problem with the notation of [5] is that the dependence of $u(z, q)$ and $R(c, q, p)$ on k is not explicit.

Finally, define $R(c, q, p)$ as the near-field planar reflection response, in direction q, of the portion of the medium below depth c, to an impulse $\delta(q-p)e^{-ik(q)z}/k(p)$, in direction p (recall directions are specified by wavenumbers). Two *inverse* Fourier transforms taking $k \to t$ and $q = k_x \to x$, as in (2.15), transform $\delta(q - p)e^{-ik(q)z}$ into the impulsive plane wave $\delta(t - z\cos\theta - x\sin\theta)$, where θ is the angle of incidence (measured from the vertical) defined by $p = k\sin\theta$. Hence $k(p)R(0, q, 0)$, computed for each k and then inverse Fourier transformed as in (2.4), is precisely the reflection response to an impulsive plane wave normally incident on the medium.

We sketch through the derivation of the invariant imbedding equations to show the similarities and differences to layer stripping. A Fourier transform of the Schrödinger equation (2.1) taking $x \to q = k_x$ yields (recall $h(z, q) = \tilde{V}(z, k_x)$)

$$\left(\frac{\partial^2}{\partial z^2} + k^2(q) - h(z, q)*\right)u(z, q) = 0 \tag{3.1}$$

where $*$ denotes convolution in q and $k^2(q) = k^2 - q^2$. Defining

$$v(z, q) = \frac{1}{2}\left(u(z, q) + \frac{\partial u}{\partial z}\frac{1}{ik(q)}\right) \qquad w(z, q) = \frac{1}{2}\left(u(z, q) - \frac{\partial u}{\partial z}\frac{1}{ik(q)}\right) \tag{3.2}$$

it can be shown [5, p 93] that $v(z, q)$ and $w(z, q)$ satisfy the coupled system (compare with (2.8))

$$\frac{d}{dz}\begin{bmatrix} v \\ w \end{bmatrix} = \begin{bmatrix} ik(q)v - (h*(v+w))/(2ik(q)) \\ -ik(q)w + (h*(v+w))/(2ik(q)) \end{bmatrix} \tag{3.3}$$

where all variables are functions of (z, q).

Now *imbed* the system (3.3) as follows. Let $A(z, c, q, p)$ and $B(z, c, q, p)$ satisfy (3.3), initialized with $A(z = c, c, q, p) = \delta(q - p)/k(p)$ and $B(z = L, c, q, p) = 0$ (the latter is a radiation condition; recall $V(x, z)$ has support in $0 < z < L$). Then

$$\begin{aligned} v(z, q) &= kA(z, 0, q, p) & w(z, q) &= kB(z, 0, q, p) \\ A(c, c, q, p) &= \delta(q - p)/k(p) & R(c, q, p) &= B(c, c, q, p). \end{aligned} \tag{3.4}$$

Furthermore, $\partial A/\partial c$ and $\partial B/\partial c$ also satisfy (3.3), but with initial conditions

$$\frac{\partial A}{\partial c}(c, c, q, p) = -\frac{\partial A}{\partial z}(c, c, q, p) \qquad \frac{\partial B}{\partial c}(L, c, q, p) = 0. \tag{3.5}$$

By superposition, the solution to (3.3) with these initial conditions is [5, p 95]

$$\frac{\partial A}{\partial c}(z, c, q, p) = -\int A(z, c, q, q')k(q')\frac{\partial A}{\partial z}(c, c, q', p)\,dq' \tag{3.6a}$$

$$\frac{\partial B}{\partial c}(z, c, q, p) = -\int B(z, c, q, q')k(q')\frac{\partial A}{\partial z}(c, c, q', p)\,dq'. \tag{3.6b}$$

We also have from the last of (3.4)

$$\frac{dR}{dc}(c, q, p) = \frac{dB}{dc}(c, c, q, p) = \frac{\partial B}{\partial z}(c, c, q, p) + \frac{\partial B}{\partial c}(c, c, q, p). \tag{3.7}$$

Finally, setting $z = c$ in (3.3), substituting into (3.6), and substituting again into (3.7) gives the following *invariant imbedding equation for R(c,q,p)*:

$$\mathrm{i}\frac{\mathrm{d}R}{\mathrm{d}c}(c,q,p) = (k + k(q))R(c,q,p) + h(c,q-p)/(2k(q)k) + \iint R(c,q,q')h(c,q'-q'')R(c,q'',p)/2\,\mathrm{d}q'\,\mathrm{d}q'' + \int (h(c,q-q')R(c,q',p)/k(q) + R(c,q,q')h(c,q'-p)/k)/2\,\mathrm{d}q'$$
$$R(L,q,p) = 0. \tag{3.8}$$

This is formaula (11*a*) in [5, p 97].

Note that (3.8) is computed recursively in decreasing c, starting at $c = L$ and ending at $c = 0$. This must be done for each p, q, and k (recall that $R(c,q,p)$ also depends on k; this dependence is not shown explicitly in (3.8) since none of the integrations are over k). Having computed $R(0,q,p)$ for all k, i.e. having computed $R(0,q,p,k)$, the inverse Fourier transform (2.4) of $kR(0,q,0,k)$ is precisely the reflection response to an impulsive plane wave normally incident on the medium. This is the scattering data used as input to the layer stripping algorithm.

3.2. Numerical implementation of invariant imbedding algorithm

Despite its apparent complexity, (3.8) can be implemented numerically in a straightforward manner by discretization similar to that used to obtain (2.11) from (2.8). Since (3.8) is already in the wavenumber domain, and the scattering potential $h(z,q)$ is known exactly, no computational instability issues arise. The integrals may be evaluated using the trapezoidal rule, and a backward difference approximation to $\mathrm{d}R/\mathrm{d}c$ used to propagate (3.8) in *decreasing* c from $c = L$ to $c = 0$.

Once again we assume a periodicity of $1/\Delta$ in the values of all functions of wavenumbers; this corresponds to the discretized functions being actual discrete values, rather than sampled values of bandlimited continuous functions. The infinite integrals in (3.6) and (3.8) become cyclic integrals (computed only over one period), so their evaluation is straightforward. The multiplication by $k + k(q)$ in (3.8) is windowed to zero for values greater than $1/(2\Delta)$, as in (2.10), and then periodically extended.

Note that it is not possible to compute the reflection response for $k = 0$ or $k(q) = 0$, due to the divisions by these in (3.8). The former can be assumed to be zero, since a non-zero DC reflection response would represent *permanent* displacement resulting from the impulsive plane wave! The latter corresponds to incidence at 90 degrees, which would not create a backscattered field in the $+z$ direction. Hence omitting these does not present a problem.

3.3. Born approximation to invariant imbedding algorithm

The invariant imbedding equation (3.8) is suggestive of a 2D version of the Riccati equation familiar in 1D scattering in layered media. The two integral terms correspond to the square term in the 1D Riccati equation. To aid in understanding (3.8), we now apply the Born approximation to (3.8), and show that the Born approximation to the layer stripping algorithm (2.14*b*) and (2.16) reconstructs the potential from the

reflection response generated by the Born approximation to the invariant imbedding equation (3.8).

As in section 2.4, we scale the potential by a small parameter ϵ, expand the wavefield and reflection response in a Taylor series in ϵ, and discard all terms of order ϵ^2 or smaller. The result is elimination of the two integrals of products terms in (3.8), leaving

$$\mathrm{i}\frac{\mathrm{d}R}{\mathrm{d}c}(c,q,p) = (k + k(q))R(c,q,p) + h(c,q-p)/(2k(q)k) \qquad R(L,q,p) = 0. \tag{3.9}$$

Since there is no longer coupling between $R(c,q,p)$ of different p, we can set $p = 0$ (normal incidence) and solve the differential equation (3.9), yielding

$$kR(z,q,k) = -\mathrm{i}\mathrm{e}^{-\mathrm{i}(k(q)+k)z}\int_L^z \frac{\hat{V}(z',q)}{2k(q)}\mathrm{e}^{\mathrm{i}(k(q)+k)z'}\,\mathrm{d}z'. \tag{3.10}$$

The factor of k multiplying $R(z,q,k)$ is present because $k(p)R(z,q,p,k)$ is the Fourier transform of the reflection response to an impulse, as discussed in the second paragraph of section 3.1. Since $p = 0$ here, we have $k(0) = k$, so $kR(z,q,k)$ is the frequency-domain reflection response to a planar impulse.

Equation (3.10) has a very clear interpretation: to form the reflection response at depth z in the Born approximation, assume the incident impulsive plane wave penetrates without being scattered to each depth z', and is then scattered by the potential $\hat{V}(z',q)$ at that depth. Then use the migration operator $\mathrm{e}^{\mathrm{i}k(q)z}$ to migrate each scattered field back to depth z independently (neglecting all coupling), and superpose the scattered fields due to each $\hat{V}(z',q)$. At the surface $z = 0$ this is clear, but it applies to any depth z.

Note that $\hat{q}(k_x,z,k)$ in (2.16) in the time domain is causal for all z (see figure 2(*b*)) while a time delay/advance $\mathrm{e}^{-\mathrm{i}kz'}$ must be included in (3.10). Also note that $\hat{V}(z',q)$ in (3.10) is scaled by $-\mathrm{i}/(2k(q))$; the reason for this will become apparent in (3.13) below.

Now consider the Born approximation to the layer stripping algorithm (2.14b) and (2.16) applied to (3.10). Taking the Fourier transform (2.4) of (2.8*a*) and using (2.16) gives

$$\hat{q}(z,q,k) = \left(\frac{\partial}{\partial z} + \mathrm{i}k\right) \qquad p(z,q,k) = \left(\frac{\partial}{\partial z} + \mathrm{i}k\right)R(z,q,k)\mathrm{e}^{\mathrm{i}k(q)z}. \tag{3.11}$$

Inserting (3.10) into (3.11) shows that the Born-approximated layer stripping algorithm computes

$$q(z,q,k) = -\mathrm{i}/(2k(q))\hat{V}(z,q)\mathrm{e}^{\mathrm{i}k(q)z} \tag{3.12}$$

from the Born-approximated scattering data. Using (2.14*b*) shows that the Born-approximated layer stripping algorithm computes

$$-2\mathcal{F}_{k\to t}^{-1}\{\hat{q}(z,q,k)\}_{t=z} = \hat{V}(z,q)\mathcal{F}_{k\to t}^{-1}\left\{\frac{\mathrm{e}^{\mathrm{i}k(q)z}}{-\mathrm{i}k(q)}\right\}_{t=z} = V(z,q) \tag{3.13}$$

so that it does indeed correctly compute the scattering potential $V(z,q)$ in the Born approximation.

4. Numerical results

4.1. Initialization

The algorithm described in section 3 was used to generate the backscattered reflection response $kR(0,q,k)$ to an impulsive plane wave for several different scattering potentials $V(x,z)$. The inverse Fourier transform (2.4) $\partial \check{R}(0,q,t)/\partial t$ of $kR(0,q,k)$ was then used to initialize the discrete layer stripping algorithm of section 2, with (recall $q = k_x$)

$$p(0,t,k_x) = \check{R}(0,q=k_x,t) \qquad q(0,t,k_x) = 2\frac{\partial}{\partial t}\check{R}(0,q=k_x,t). \tag{4.1}$$

The latter initial condition comes from (2.7*a*) and the fact that $\check{R}(z,q,t) = \check{R}(0,q,t+z)$ in the homogeneous overlying half-space $z < 0$, since $R(0,q,t)$ is a *backscattered* (i.e. upward-traveling) wave. Note that the sample applications of section 2 would require different initial conditions.

4.2. Forward problem versus inverse problem algorithms

The invariant imbedding algorithm was used to generate the forward data so that the layer stripping inverse problem algorithm would not simply run the computations of the forward problem algorithm backwards. Although the two algorithms must of course be mathematically equivalent, since they are both 'exact', they are derived from different mathematical principles.

Some specific differences between the forward problem (invariant imbedding) algorithm (FPA) and the inverse problem (layer stripping) algorithm (IPA) are as follows.

(i) The FPA propagates the reflection coefficient at depth $R(c,q,p,k)$. The IPA propagates the field and field gradient $p(z,t,k_x)$ and $q(z,t,k_x)$. Note that $R \neq q/p$, since R is the ratio of downgoing and upgoing *waves*, not field quantities.

(ii) The FPA operates in the k (frequency) domain, while the IPA operates in the t (time) domain.

(iii) The FPA computes $R(c,q,p,k)$ for all c,q,p,k, while the IPA is initialized using $kR(0,q,0,k)$, a slice of the FPA function. Note in the FPA (3.8) the integrals over q' and the differences $q'-q''$; these clearly have no counterpart in the IPA.

(iv) The FPA propagates (3.8), which can be viewed as a 2D generalization of the Riccati equation familiar in 1D inverse scattering. The IPA propagates the coupled system (2.8); note that this differs from the coupled system (3.3) used to derive (3.8).

(v) While both algorithms are discretized in depth, the FBP results did not vary significantly with mesh size, so this should not be an issue. The IBP results also did not vary significantly with mesh size, and gave good results at several different resolutions (see below).

These differences make it clear that errors are not cancelling out algebraically between the FPA and the IPA, i.e., the IPA is not effectively running the FPA computations backwards.

4.3. Summary of results

The numerical performance of the layer stripping algorithm was studied under a variety of conditions. The results may be summarized as follows.

(i) The layer stripping algorithm successfully reconstructed the potential in the absence of noise. The only difficulty was due to the smoothing of the transverse derivative, which slightly smoothed very sharp variations in the lateral direction.

(ii) The layer stripping algorithm continued to work well when a small amount of Gaussian random noise was added to the reflection response. The reconstructed potential was slightly degraded, of course, but the amount of degradation seemed to vary smoothly with the amount of noise—a slight increase in noise level did not vastly degrade the reconstructed potential.

(iii) The layer stripping algorithm reconstructions were superior to those using the Born approximation (as specified in section 2.4 above), in that the Born approximation treated multiple scattering events as additional single scattering events, resulting in errors in the reconstructed potential, particularly for large z. This effect was more pronounced when the potential had numerically large values; for small potentials $V(n\Delta,k\Delta)\Delta \ll 1$ the Born approximation worked quite well. This was as expected; multiple scattering involves products of potentials, and multiplying small values results in even smaller values.

(iv) The performance of the algorithm seemed to vary little with the size of the discretization length Δ, provided that the same Δ was used in the discretized invariant imbedding algorithm. This suggests there may be a close relation between the discretized versions of these algorithms. Coarse grid reconstructions seemed to be merely undersampled versions of the fine grid reconstructions; the basic features of the reconstructions were identical.

We illustrate these points with some numerical examples below. It should be noted that the following is only a representative and illustrative sample of our results; the above conclusions are not based merely on the results below. Unless otherwise specified, all examples used $\Delta = 1/32$, $L = L_x = 1/2$, and $\Delta_k = 1$. The 3D plots are depicting 2D functions $V(x,z)$; they do not represent objects buried in a homogeneous surrounding medium.

4.4. Comparison with the Born approximation

The potential $V(x,z)$ is shown in figure 3(*a*). Note that this is a smooth, rounded potential having compact support in both x and z.

The reconstructed potential using the Born approximation is shown in figure 3(*b*). Although figure 3(*b*) superficially seems to be identical to figure 3(*a*), study carefully the deepest part of the reconstructed $V(x,z)$. The original $V(x,z)$ is zero for $z > 24/32$, while the Born-reconstructed $V(x,z)$ does not become zero until $z > 26/32$; it has a 'tail'. This 'tail' is caused by multiple scattering that is interpreted under the Born approximation as primary scattering due to an additional non-zero portion of the scattering potential; actually, there is no such portion.

The reconstructed potential using the layer stripping algorithm is shown in figure 3(*c*). This reconstruction has no 'tail'; the multiple scattering that produces it has been accounted for in the algorithm and eliminated. The reconstruction is almost perfect.

A different potential is shown in figure 4(*a*). Note that this potential function is constant over a central 'plateau', and then drops off rapidly to zero.

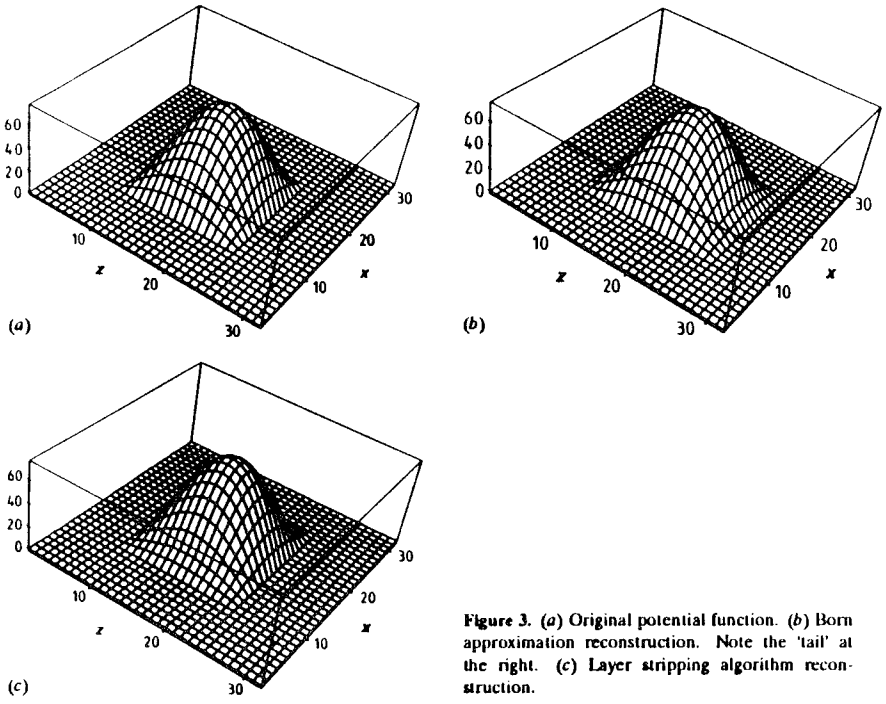


Figure 3. (*a*) Original potential function. (*b*) Born approximation reconstruction. Note the 'tail' at the right. (*c*) Layer stripping algorithm reconstruction.

The reconstructed potential using the Born approximation is shown in figure 4(*b*). Note again the presence of a 'tail' at its deepest part, while there is no 'tail' at its shallowest part, since multiple scattering has not yet had *time* to occur in this part of the time-domain impulse response (the lack of symmetry in z is apparent if one looks at the figure as a whole). Also note the problems in reconstructing the lateral edges of the potential function; the central 'plateau' is much smaller than it should be.

The reconstructed potential using the layer stripping algorithm is shown in figure 4(*c*). Again the 'tail' caused by multiple scattering has been eliminated. However, the shallowest and deepest edges of the 'plateau' have been rounded off slightly. Since this is symmetric between the shallowest and deepest parts, it is not due to multiple scattering. We attribute it to smoothing in the transverse derivative.

4.5. *Effects of additive noise*

The potential $V(x, z)$ used in figure 3 was scaled as shown in figure 5(*a*), and Gaussian random noise was added to the reflection response $R(0, q, k)$. The signal-to-noise ratio, computed as the square root of the sum of the squares of the discrete signal values divided by the square root of the sum of the squares of the discrete noise values, was found to be 36 dB for one run and 18 dB for another (to get power SNR these values should be doubled). Note that any powers of Δ and numbers of points being averaged will cancel in this ratio.

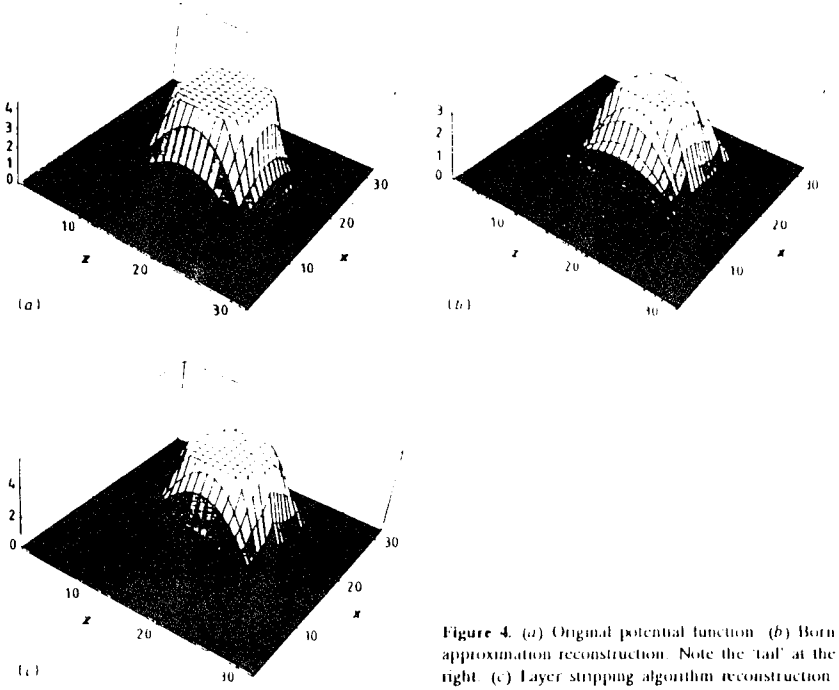


Figure 4. (*a*) Original potential function. (*b*) Born approximation reconstruction. Note the 'tail' at the right. (*c*) Layer stripping algorithm reconstruction.

The reconstructions at 36 dB are virtually perfect; in fact, the reconstructions shown in figure 3 are actually these reconstructions. The reconstructions at 18 dB are shown in figure 5(*b*) using the Born approximation and figure 5(*c*) using the layer stripping algorithm. Note that even in these noisy reconstructions the 'tail' is still a significant feature in the Born approximation reconstruction, while the layer stripping reconstruction has correctly removed the 'tail'.

To see the degradation of the layer stripping algorithm in the presence of increasing amounts of noise added to the reflection response, study figure 6. Figure 6(*a*) shows the original potential function, which is the same as figure 4(*a*). Figure 6(*b*) shows a noisy reconstruction of the potential function shown in figure 6(*a*), and in figure 6(*c*) the signal-to-noise ratio has been reduced by a factor of four. The increasing degradation of the reconstruction is obvious, but the layer stripping algorithm does not fall apart even in large amounts of additive noise.

A similar study is carried out for a different potential function in figure 7. Figure 7(*a*) shows the original potential function, and figures 7(*b*) and 7(*c*) correspond to figures 6(*b*) and 6(*c*). The only notable feature of the layer stripping reconstructions is the slight (one pixel wide) 'shelf' induced by the smoothed transverse derivative (this is discussed in more detail below); otherwise, the reconstructed potential smoothly degrades with increasing noise.

Note the presence of the sharp ridge along the line $z = 0$ in figures 6 and 7. This ridge is due to the non-zero mean of the noise being added to $R(0, q, k)$. When

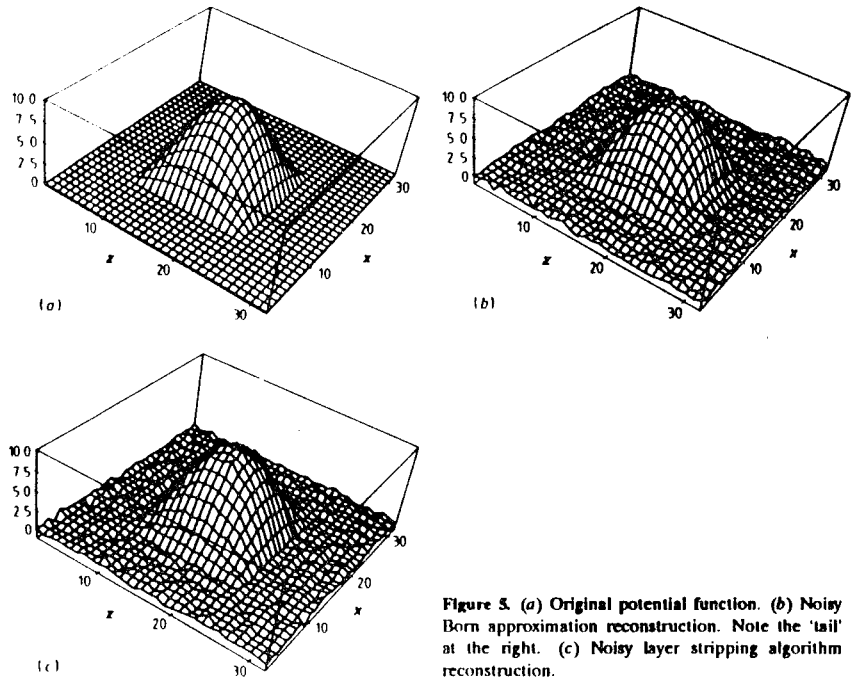


Figure 5. (*a*) **Original potential function.** (*b*) Noisy Born approximation reconstruction. Note the 'tail' at the right. (c) Noisy layer stripping algorithm reconstruction.

$V(x, z)$ is computed by taking the inverse Fourier transform (2.9), this non-zero mean, a constant in the Fourier wavenumber k domain, becomes an impulse in the spatial z domain. This impulse is the ridge.

4.6. Discussion of numerical stability with noise

The smooth degradation of the reconstructed potential with increasing noise levels might seem surprising, since the inverse scattering problem is known to be ill-conditioned. The reason for this is that multiple scattering has a relatively small (compared to single scattering) effect, so that the Born approximation result will be approximately the same as the layer stripping result. The Born approximation is linear, so that any noise added to the reflection response will produce an addition to the reconstructed potential whose strength is directly proportional to the noise strength (halving the noise will halve the addition); hence the Born-reconstructed potential will degrade smoothly, and it is not surprising that the reconstructed potential from layer stripping also degrades smoothly.

This heuristic argument should not be taken too far; in the 1D case, it is well known that large noise levels can cause severe problems in layer stripping algorithms, and indeed in *any* 'exact' method. The reason for this is *not* numerical instability, as is commonly believed; the 1D layer stripping algorithm is identical to the Schur algorithm (see [8]), which is *known to be numerically stable*.

The reason that 1D layer stripping algorithms can give unstable results when they

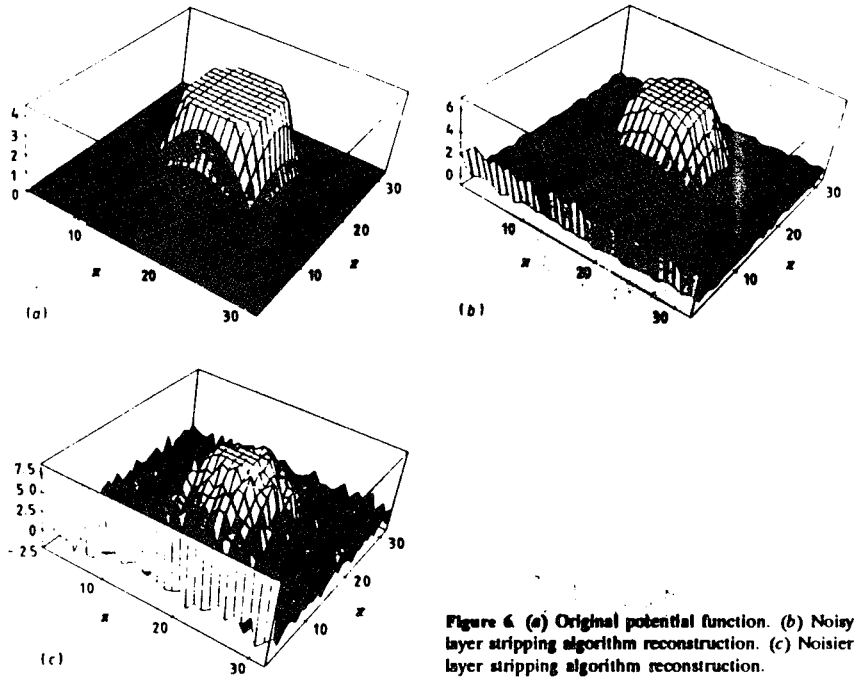


Figure 6. (*a*) **Original potential function.** (*b*) Noisy layer stripping algorithm reconstruction. (*c*) Noisier layer stripping algorithm reconstruction.

are applied to noisy reflection data is as follows. It is well known that the free-surface reflection response of a 1D layered medium to an impulsive plane wave below the surface is one side of the autocorrelation of its transmission response; hence it must be positive semi-definite. Noise added to the reflection response can make the two-sided response (the reflection response added to its time reversal) become non-positive semi-definite, in which case it is no longer the reflection response to *any* layered medium. The problem is now ill-posed, in the sense of having no solution; it is not surprising that the layer stripping algorithms become unstable.

However, small amounts of additive noise will not cause the reflection response to become non-positive semi-definite; as long as this is true, the layer stripping algorithms will behave well numerically. Our results in this paper suggest that a a similar situation is present in the 2D inverse scattering problem considered here; this is a topic of current research.

4.7. Smoothed reconstructions due to smoothed transverse derivative

The smoothing in the transverse derivative incurred by using the clipped filter (2.10) causes a slight but noticeable smoothing of $V(x, z)$ along the x direction. This was manifested in the reconstructions in figure 7 by the 'shelf' that appeared at the ends of the reconstructed potential function. Another example of this is illustrated in figure 8. Figure 8(*a*) shows the original potential function, which was produced by taking the potential function of figure 7(*a*) and adding random noise to it. The

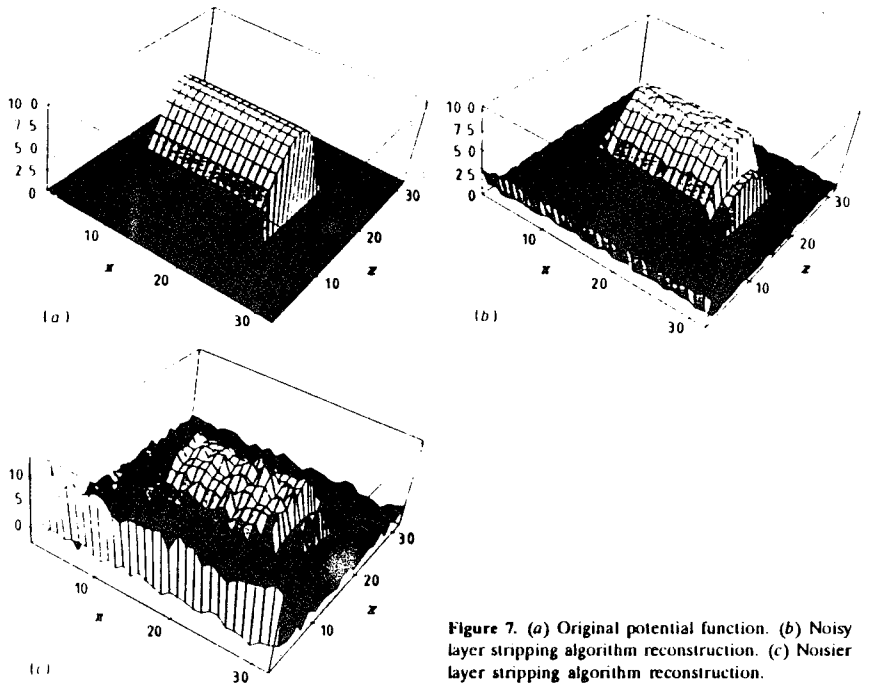


Figure 7. (*a*) Original potential function. (*b*) Noisy layer stripping algorithm reconstruction. (*c*) Noisier layer stripping algorithm reconstruction.

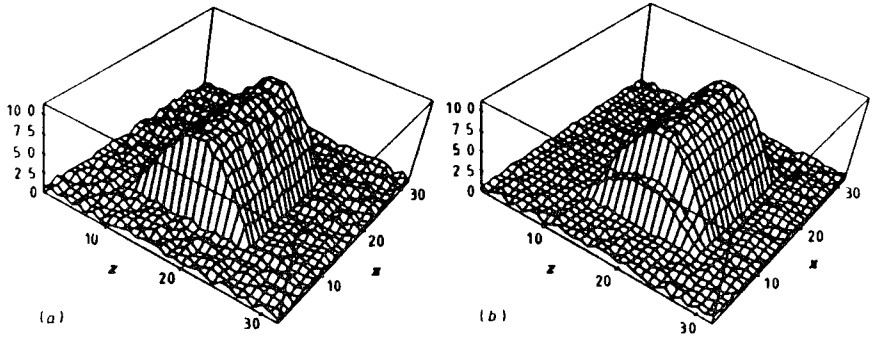


Figure 8. (*a*) Original 'noisy' potential function. (*b*) Layer stripping algorithm reconstruction.

idea here is that in real life potential functions will not have simple analytic forms; they will be complicated functions. Hence figure 8(*a*) is closer to a realistic potential function.

The reconstructed potential from the layer stripping algorithm is shown in figure 8(*b*). Note again the one-pixel-wide 'shelf' at each of the two flat ends of the potential function. We attribute this to the smoothing of the transverse derivative in

the layer stripping algorithm; unable to reconstruct the sharp jump from zero, the algorithm provides a laterally smoothed reconstruction in which the reconstructed potential takes two smaller lateral jumps instead of a single large jump. Note that the 'shelf' is half the height of the jump in x at each depth z.

Also note in figure 8(*b*) that the 'noisy' part of the potential in figure 8(*a*) has been noticeably smoothed. This again seems to be due to the smoothing in the transverse derivative; note that the reconstructed potential is 'rougher' in the z direction (for which there is no smoothing) than in the x direction (in which there is smoothing). This smoothing effect should be taken into consideration in potentials reconstructed using layer stripping algorithms.

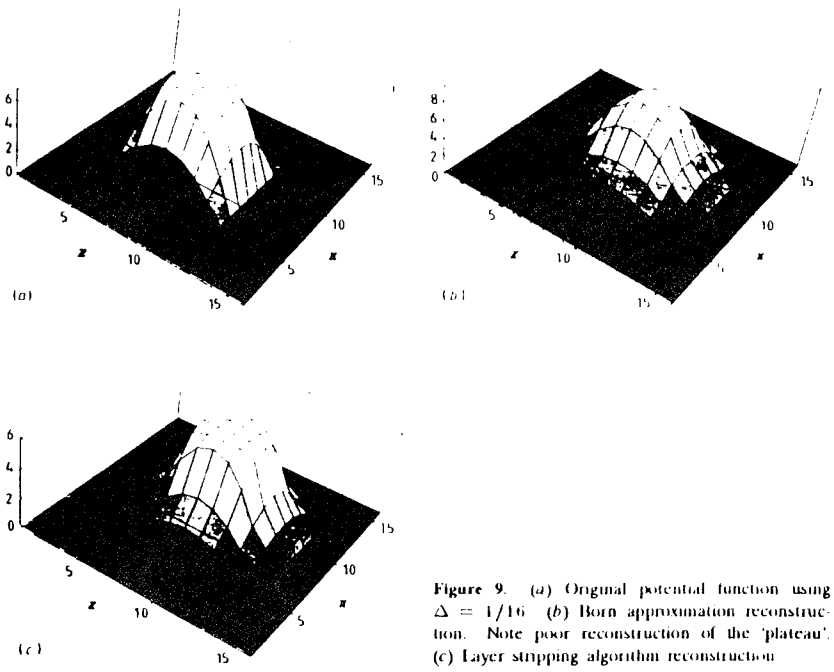


Figure 9. (*a*) Original potential function using $\Delta = 1/16$. (*b*) Born approximation reconstruction. Note poor reconstruction of the 'plateau'. (*c*) Layer stripping algorithm reconstruction.

4.8. *Effect of discretization length* Δ

The above numerical runs all used $\Delta = 1/32$. Results for a larger $\Delta = 1/16$ are shown in figure 9. Figure 9(*a*) shows the original potential, which is an undersampled version of the potential in figure 4(*a*). The reconstructed potential using the Born approximation is shown in figure 9(*b*). Note how poorly the Born approximation reconstructs the central 'plateau' of the potential function. The reconstructed potential using the layer stripping algorithm is shown in figure 9(*c*). Although the central 'plateau' is reconstructed quite well, the potential function as a whole is spread out one pixel in each direction. This shows that while the invariant imbedding and

layer stripping algorithms are clearly closely connected, the discretized layer stripping algorithm is *not* merely running the invariant imbedding algorithm backwards.

Results for a smaller $\Delta = 1/64$ are shown in figure 10. Figure 10(a) shows the original potential, which is a more finely sampled version of the potential in figure 4(a). The reconstructed potential using the layer stripping algorithm is shown in figure 10(b). The reconstruction is almost perfect—even the lateral smoothing caused by the smoothed transverse derivative is not apparent. This is due to the fact that although $\Delta_k = 1$, the maximum value of k_x is now 32 instead of 16; the smoothing starts at a much higher wavenumber. A very close comparison of figures 10(a) and 10(b) show that the reconstruction is not quite perfect; the reconstructed potential is still spread out one pixel in each direction. But this effect is virtually negligible on this scale.

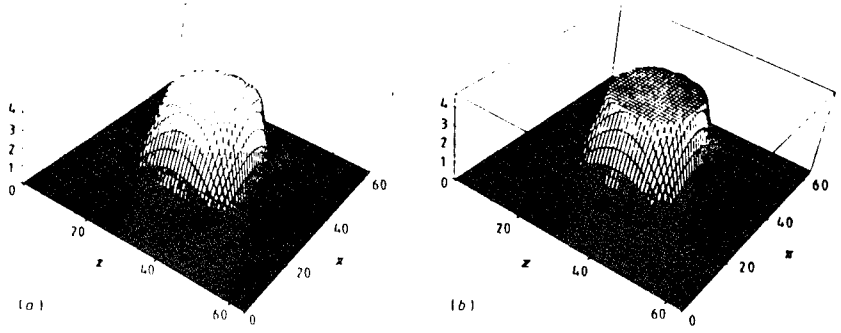


Figure 10. (a) Original potential function using $\Delta = 1/64$. (b) Layer stripping algorithm reconstruction.

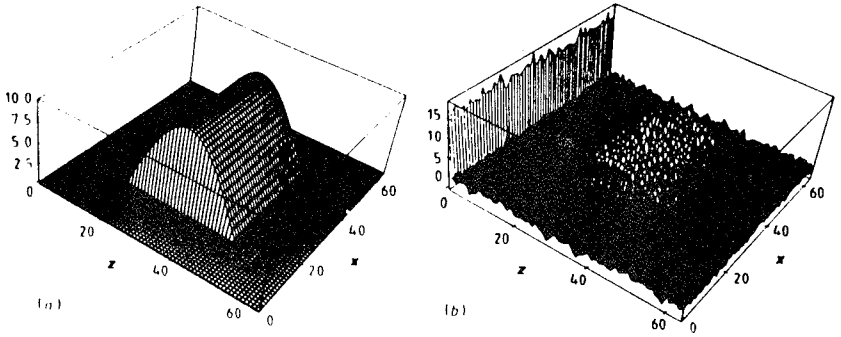


Figure 11. (a) Original potential function using $\Delta = 1/64$. (b) Noisy layer stripping algorithm reconstruction.

One final example combines a smaller Δ, additive noise in the reflection response, and smoothed reconstruction. Figure 11(a) shows the original potential, which is a more finely sampled version of the potential in figure 7(a). Random noise was added to the reflection data, at a signal-to-noise ratio of 15.7 dB. The reconstructed potential

using the layer stripping algorithm is shown in figure 11(b). All the features discussed in section 4.3 are again present in figure 11(b). These include the 'shelf', still one pixel wide, the ridge along the line $z = 0$, and the main shape of the potential function still visible in the noise. This shows that these effects occur at different discretization lengths, and indeed may be endemic to layer stripping reconstructions with noise for any Δ.

5. Conclusion

The numerical performance of the 2D layer stripping algorithm of [4] has been studied for the first time. This represents the first numerical implementation of an 'exact' non-iterative inverse scattering algorithm that includes the effects of *all* multiple scattering and diffraction effects. The forward scattering data were generated using the invariant imbedding algorithm of [5]. The results indicated that layer stripping is a viable technique for solving 2D Schrödinger equation inverse potential problems, for which two applications were briefly reviewed.

Two particularly important results were that: (1) the 'exact' reconstructions using the layer stripping algorithm are a noticeable improvement over the Born approximation reconstructions; and (2) small amounts of additive noise in the reflection response do not cause numerical instability in the layer stripping algorithm. The results were illustrated using several numerical examples. It was also shown for the first time that the Born approximation to the layer stripping algorithm reconstructs the scattering potential from the reflection response generated by the Born approximation to the invariant imbedding algorithm of [5].

Acknowledgments

It is a pleasure to acknowledge the help of Ms Susan Wei, who helped with the preliminary stages of this research. This research was supported by the Office of Naval Research under grant # N00014-90-J-1897.

References

[1] Coen S, Cheney M and Weglein A 1984 Velocity and density of a two dimensional acoustic medium from point source surface data *J. Math. Phys.* **25** 1857–60

[2] Yagle A E 1987 A layer stripping fast algorithm for the two-dimensional direct current inverse resistivity problem *IEEE Trans. Geosci. Rem. Sensing* **GE-25** 558–63

[3] Cheney M 1984 Inverse scattering in dimension two *J. Math. Phys.* **25** 94–102

[4] Yagle A E and Levy B C 1986 Layer stripping solutions of multidimensional inverse scattering problems *J. Math. Phys.* **27** 1701–10

[5] Wilcox R 1970 Wave propagation through longitudinally and transversally inhomogeneous slabs—I *Invariant Imbedding* ed R E Bellman and E D Denman (New York: Springer)

[6] Cheney M 1990 Stability analysis of the Yagle–Levy multidimensional inverse scattering algorithm *Preprint* Department of Mathematical Sciences, Rensselaer Polytechnic Institute, Troy, NY

[7] Snieder R 1990 The role of the Born approximation in nonlinear inversion *Inverse Problems* **6** 247–66

[8] Bruckstein A M, Levy B C and Kailath T 1985 Differential methods in inverse scattering *SIAM J. Appl. Math.* **45** 312–35

Numerical Performance of Layer Stripping Algorithms for Two-Dimensional Inverse Scattering Problems

Andrew E. Yagle and Poovendran Raadhakrishnan
Dept. of Electrical Engineering and Computer Science
The University of Michigan, Ann Arbor, MI 48109-2122
Revised April 1992

Abstract

Numerical results of implementing a two-dimensional layer stripping algorithm to solve the two-dimensional Schrodinger equation inverse potential problem are presented and discussed. This is the first exact (all multiple scattering and diffraction effects are included) numerical solution of a multi-dimensional Schrodinger equation inverse potential problem, excluding optimization-based approaches. The results are as follows: (1) the layer stripping algorithm successfully reconstructed the potential from scattering data measured on a plane (as it would be in many applications); (2) the algorithm avoids multiple scattering errors present in Born approximation reconstructions; and (3) the algorithm is insensitive to small amounts of noise in the scattering data. Simplifications of layer stripping and invariant imbedding algorithms under the Born approximation are also discussed.

ICP Classification Numbers: 0270, 0380.

1. Introduction

The inverse scattering problem for the Schrodinger equation in two dimensions with a time-independent, local, non-circularly symmetric potential has many applications. Two of these applications are as follows: (1) reconstruction of a three-dimensional (3-D) acoustic medium with density and wave speed varying in two dimensions (2-D), from surface measurements of the steady-state medium displacement response to a harmonic line source [1]; and (2) reconstruction of a 3-D electrical medium with resistivity varying in 2-D from surface measurements of the potential resulting from a line DC current source [2]. Both of these applications are quickly reviewed below in Section 2.1.

Two major approaches for exact solution of the 2-D Schrodinger equation inverse potential problem have been proposed. The first is the 2-D version of the Gel'fand-Levitan and Marchenko integral equation methods [3]. The other is the 2-D version of the layer stripping differential methods [4]. Here "exact" means that all diffraction and multiple scattering effects are included in the mathematical solution; errors in the solution will arise solely due to purely numerical effects such as discretization and roundoff. Hence all methods based on the Born (single-scattering) approximation are excluded here, since such methods, and their modifications, do not take into account *all* multiple scattering effects. In Section 2.4 we discuss how the Born approximation applies to the algorithm of [4]. No numerical implementation of the methods of either [3] or [4] has previously been reported.

This paper presents the results of the first numerical implementation of the 2-D version of the layer-stripping algorithm of [4]. It is thus the first *exact* (as defined above) numerical solution of a multi-dimensional Schrodinger equation inverse potential problem. Note that optimization-based approaches minimize (or maximize) some criterion; thus they are not in the spirit of the approach considered here. Although only reconstruction of the Schrodinger scattering potential is considered here, direct application to specific inverse scattering problems, as in [1] and [2], would be possible.

This paper is organized as follows. In Section 2 the 2-D Schrodinger equation inverse potential problem is formulated, two applications are noted, the layer stripping algorithm of [4] is reviewed, details of its numerical implementation are discussed, and its simplifi-

cation under the Born approximation is discussed. In Section 3 the invariant imbedding algorithm of [5] used to generate the scattering data is reviewed, and details of its numerical implementation are discussed. We also discuss its simplification under the Born approximation, and show analytically that the Born-simplified layer stripping algorithm successfully inverts the Born-simplified invariant imbedding algorithm scattering data. Although the latter result is new, it is intended primarily to give some feel for the algorithms of [4] and [5].

Section 4 summarizes the numerical results, and presents some illustrative examples. Issues illustrated include: (1) errors in reconstructed potentials using the Born approximation, which are eliminated using the "exact" layer stripping algorithm; (2) effects on reconstructed potentials of various amounts of noise in the data; (3) effects on reconstructed potentials of regularization of transverse derivatives in the layer stripping algorithm; and (4) effects of choosing various discretization lengths in the layer stripping algorithm. Section 5 concludes with a summary.

2. Two-Dimensional Layer Stripping Algorithm

2.1 Problem Formulation and Applications

The 2-D inverse scattering problem considered in this paper is as follows. The problem is defined in 2-D (x, z) space, where x is lateral position and z is depth, increasing downward from the surface $z = 0$. The wave field $\hat{p}(x, z, k)$ satisfies the 2-D Schrodinger equation

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial z^2} + k^2 - V(x, z)\right) \hat{p}(x, z, k) = 0, \tag{2.1}$$

where the potential $V(x, z)$ is real-valued, smooth, and has support in z in the interval $0 < z < L$. It is also assumed that $V(x, z)$ does not induce bound states; a sufficient condition for this is for $V(x, z)$ to be non-negative.

The medium is probed by an impulsive plane wave e^{-ikz}, which passes through the surface $z = 0$ at time $t = 0$ and induces scattering by $V(x, z)$ for $t > 0$. The scattering data consists of measurements of the wave field $\hat{p}(x, z^*, k)$ and its gradient $\frac{\partial \hat{p}(x, z^*, k)}{\partial z}$ for some z^* in the homogeneous half-space $z \geq 0$. For convenience, we assume measurements

are taken at the surface $z^* = 0$, as they would be in the applications to follow. The inverse scattering experiment is illustrated in Fig. 1.

We now quickly review two applications of this problem. First, consider the problem of reconstructing a 3-D inhomogeneous acoustic medium whose density $\rho(x,z)$ and wave speed $c(x,z)$ are smooth functions of depth z and lateral position x. The medium is bounded by a free (pressure-release) surface $z = 0$. The density ρ_0 and wave speed c_0 for $z < 0$ and $z \to \infty$ are known. The medium is probed with cylindrical harmonic waves, at two frequencies ω_1 and ω_2, from a harmonic line source extending along the x-axis, and the sinusoidal steady-state vertical acceleration $\hat{a}(x,y,z=0;\omega_i)$ of the medium at the free surface $z = 0$ is measured. The goal is to reconstruct $\rho(x,z)$ and $c(x,z)$ from the measurements $\hat{a}(x,y,z=0;\omega_i), i = 1,2$.

This problem can be formulated as a 2-D Schrodinger equation inverse potential problem by Fourier transforming the basic acoustic equations with respect to time and the other lateral variable y. Details are given in both [1] and [4]. Here we merely note that in the Schrodinger equation (2.1) the wave field $\hat{p}(x,z,k)$ is pressure divided by $\rho(x,z)^{1/2}$, the wavenumber $k^2 = \omega_i^2/c_0^2 - k_y^2$, and the potential $V(x,z;\omega_i)$ is

$$V(x,z;\omega_i) = \left(\frac{\omega_i^2}{c_0^2}\right)\left(1 - \frac{c_0^2}{c(x,z)^2}\right) + \rho(x,z)^{1/2}\nabla^2(\rho(x,z)^{-1/2}). \tag{2.2}$$

It is clear that performing this experiment for two different frequencies $\omega_i, i = 1,2$ will allow $\rho(x,z)$ and $c(x,z)$ to be computed from (2.2). The wave field is zero at the free surface $z = 0$; its gradient is the medium acceleration $\rho(x,0)^{1/2}\hat{a}(x,y,z=0;\omega_i), i = 1,2$.

The second application is the inverse resistivity problem of reconstructing a 3-D inhomogeneous electrical medium whose resistivity $\rho(x,z)$ is a smooth function of x and z over a bounded region. The medium is probed with current from a line DC current source extending along the x-axis, and the electrical potential $v(x,y,z=0)$ induced on the surface $z = 0$, assumed to be a perfect insulator, is measured. The goal is to reconstruct the resistivity $\rho(x,z)$ from the measurements of electrical potential $v(x,y,z=0)$. Note that for both applications, the response to a *line* source may be found by superposition of the responses due to *point* sources along the x-axis.

This problem can be formulated as a 2-D Schrodinger equation inverse potential

problem by Fourier transforming Ohm's and Kirchoff's current laws with respect to the other lateral variable y. Details are given in [5]. Here we merely note that in the Schrodinger equation (2.1) the wave field $\hat{p}(x,z,k)$ is now the inverse Laplace transform of the Fourier transform of electrical potential divided by $\rho(x,z)^{1/2}$, and the scattering potential $V(x,z) = \rho(x,z)^{1/2}\nabla^2(\rho(x,z)^{-1/2})$.

2.2 The 2-D Layer Stripping Algorithm

The layer stripping algorithm for solving the 2-D Schrodinger equation inverse potential problem is derived as follows [4]. Taking the inverse Fourier transform of (2.1) with respect to k yields

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial z^2} - \frac{\partial^2}{\partial t^2} - V(x,z)\right)\check{p}(x,z,t) = 0 \tag{2.3}$$

where

$$\check{p}(x,z,t) = \frac{1}{2\pi}\int_{-\infty}^{\infty}\hat{p}(x,z,k)e^{ikt}dk. \tag{2.4}$$

Eq. (2.3) can be written as the coupled system

$$\left(\frac{\partial}{\partial z} + \frac{\partial}{\partial t}\right)\check{p}(x,z,t) = \check{q}(x,z,t) \tag{2.5a}$$

$$\left(\frac{\partial}{\partial z} - \frac{\partial}{\partial t}\right)\check{q}(x,z,t) = \left(V(x,z) - \frac{\partial^2}{\partial x^2}\right)\check{p}(x,z,t) \tag{2.5b}$$

From causality and the form of (2.5a), $\check{p}(x,z,t)$ and $\check{q}(x,z,t)$ have the forms

$$\check{p}(x,z,t) = \delta(t-z) + \tilde{p}(x,z,t)1(t-z) \tag{2.6a}$$

$$\check{q}(x,z,t) = \tilde{q}(x,z,t)1(t-z), \tag{2.6b}$$

where $\tilde{p}$ and $\tilde{q}$ are the smooth parts of $\check{p}$ and $\check{q}$, respectively, and $1(\cdot)$ is the unit step or Heaviside function.

Inserting (2.6) in (2.5) and equating coefficients of $\delta(t-z)$ (propagation of singularities argument) yields

$$\left(\frac{\partial}{\partial z} + \frac{\partial}{\partial t}\right)\tilde{p}(x,z,t) = \tilde{q}(x,z,t) \tag{2.7a}$$

$$\left(\frac{\partial}{\partial z} - \frac{\partial}{\partial t}\right)\tilde{q}(x,z,t) = \left(V(x,z) - \frac{\partial^2}{\partial x^2}\right)\tilde{p}(x,z,t) \tag{2.7b}$$

$$V(x,z) = -2\tilde{q}(x,z,t=z^+). \tag{2.7c}$$

Equations (2.7) constitute the *basic 2-D layer stripping equations*: Starting with measured $\tilde{p}(x,0,t)$ and $\tilde{q}(x,0,t)$ (the gradient of the wave field is required for the latter), propagate (2.7) recursively in increasing depth z, reconstructing $V(x,z)$ as the algorithm proceeds. The coupled equations (2.7a) and (2.7b) include ***all*** multiple scattering and diffraction effects, since they are equivalent to the Schrodinger equation (2.1) in the time domain. The potential may be reconstructed using (2.7c) since (2.7) is implemented at the wave front $t = z$; by time causality there has been no time for multiple scattering to occur yet.

Some advantages of using layer stripping algorithms are as follows:

1. Only backscattered data from one direction of probing is required. Integral equation methods [3] require the scattering amplitude, which is the far-field response in ***all*** directions to an incident impulsive plane wave in ***each*** possible direction. In the applications noted above, this is unrealistic; it also runs the risk of inconsistent data;
2. The amount of computation required is much less than the amount required to solve the integral equations of [3]. The layer stripping algorithm can be viewed as a fast algorithm solution of these integral equations which exploits the Hankel structure in the kernel of the generalized Marchenko integral equation of [3];
3. *All* multiple scattering and diffraction effects are included, unlike methods such as distorted-wave Born approximation which only account for some of these effects.

Two disadvantages of layer stripping algorithms are as follows:

1. It is not clear how to incorporate the effects of bound states (roughly, square-integrable solutions to the Schrodinger equation with negative energy); unlike the approach of [3];
2. The lateral derivative $\frac{\partial^2}{\partial x^2}$ in (2.7b) can be expected to induce numerical instability.

2.3 Numerical Implementation of the 2-D Layer Stripping Algorithm

The second disadvantage can be removed as follows. Take the Fourier transform of

(2.7) with respect to x. The result is

$$\left(\frac{\partial}{\partial z}+\frac{\partial}{\partial t}\right)p(z,t,k_x)=q(z,t,k_x) \tag{2.8a}$$

$$\left(\frac{\partial}{\partial z}-\frac{\partial}{\partial t}\right)q(z,t,k_x)=k_x^2p(z,t,k_x)+\hat{V}(z,k_x)*p(z,t,k_x) \tag{2.8b}$$

$$\hat{V}(z,k_x)=-2q(z,t=z^+,k_x) \tag{2.8c}$$

where $*$ denotes convolution in k_x,

$$p(z,t,k_x)=\int_{-\infty}^{\infty}\tilde{p}(x,z,t)e^{-ik_xx}dx \tag{2.9}$$

and $q(z,t,k_x)$ and $\hat{V}(z,k_x)$ are defined similarly.

The multiplication by k_x^2 in (2.8b) will induce numerical instability. This may be avoided by replacing the multiplication by k_x^2 in (2.8b) with multiplication by the clipped filter

$$H(k_x)=\begin{cases}k_x^2, & \text{if } |k_x|<K;\\ 0, & \text{otherwise}\end{cases} \tag{2.10}$$

for some cutoff wavenumber K. This is reminiscent of the clipped filter used in the filtered back-projection procedure for inverting the Radon transform. In practice, the discontinuities in (2.10) at $|k_x|=K$ would be replaced by a smooth window to zero; a Hanning (raised cosine) window was used in the numerical simulations presented later.

We now discretize depth $z=n\Delta$ and time $t=j\Delta$ to integer multiples of some discretization length Δ. Since the wave speed in (2.1) is unity, depth and time have the same Δ. Lateral position x would also use the same Δ; but wavenumber $k_x=k\Delta_k$ must use for Δ_k half the reciprocal of the total lateral extent of interest; e.g., if the potential has finite support $-L_x/2<x<L_x/2$ in x, L_x would be the lateral extent of interest. Note that Δ and Δ_k have reciprocal units.

Using forward difference approximations to the partial derivatives then yields

$$p((n+1)\Delta,(j+1)\Delta,k\Delta_k)=p(n\Delta,j\Delta,k\Delta_k)+q(n\Delta,j\Delta,k\Delta_k)\Delta \tag{2.11a}$$

$$q((n+1)\Delta,(j-1)\Delta,k\Delta_k)=q(n\Delta,j\Delta,k\Delta_k)+H(k\Delta_k)p(n\Delta,j\Delta,k\Delta_k)\Delta$$

$$+ \sum_{m=-\infty}^{\infty} \hat{V}((k-m)\Delta_k)p(n\Delta, j\Delta, m\Delta_k)\Delta\Delta_k \qquad (2.11b)$$

$$V((n+1)\Delta, k\Delta_k) = -2q((n+1)\Delta, (n+1)\Delta, k\Delta_k). \qquad (2.11c)$$

Equations (2.11) constitute the numerical implementation of the 2-D layer stripping algorithm. The update patterns are illustrated in Figs. 2; note that by time causality $p(z,t,k_x)$ and $q(z,t,k_x)$ are zero for $t < z$.

Cheney [6] has shown that the modification (2.10) stabilizes the layer stripping algorithm (2.11), in the following sense. Define the norm

$$||f(k_x,t)|| = \sup_{t>0} \int_{-\infty}^{\infty} |f(k_x,t)|dk_x. \qquad (2.12)$$

Input two different sets of bounded initial data $p_i(k_x,0,t), q_i(k_x,0,t), i = 1,2$ into the discretized algorithm (2.11), resulting in two different reconstructed potentials $\hat{V}_i(k_x,z), i = 1,2$. Let $||p_i(k_x,0,t)|| < K'$ and $||q_i(k_x,0,t)|| < K'$ for some K'. Then for $z = n\Delta$ we have

$$\sup_{k_x}[\hat{V}_1(k_x,z) - \hat{V}_2(k_x,z)] \leq K_1(z)||p_1(k_x,0,t) - p_2(k_x,0,t)|| + K_2(z)||q_1(k_x,0,t) - q_2(k_x,0,t)||, \qquad (2.13)$$

where $K_1(z)$ and $K_2(z)$ are polynomials in n, Δ, K, and K'.

The discretized system (2.11) can be implemented as is. However, its spectral properties are worth examining. It might seem as though we can regard the discretized functions $p(n\Delta, j\Delta, k\Delta)$, etc. as merely sampled versions of the continuous functions $p(z,t,k_x)$, etc., provided the latter are bandlimited and sampling is performed above the Nyquist rate. However, the nonlinear product in (2.7b) becomes the convolution in k_x in (2.8b) and (2.11b); the wavenumbers become mixed. Indeed, even if the inverse potential problem is regularized by assuming that $\hat{V}(z,k_x)$ is bandlimited in z and zero for $|k_x| > K$ for some K, it is clear that $p(z,t,k_x)$, etc. will NOT have similar properties. Imposing a bandlimited condition at each recursion will lead to errors, since the missing high wavenumbers will cause errors at low wavenumbers due to the wavenumber mixing. This leads to the question of what the discretized $p(n\Delta, j\Delta, k\Delta_k)$, etc. mean, and how the convolutions in k_x should be performed. It should be noted that similar questions arise in integral equation methods.

One possible interpretation is to perform a periodic extension in k of all quantities in (2.11). The period in k should be $1/\Delta_k$; K in (2.10) should then be half this. It is clear by induction that if all quantities at depth $n\Delta$ are periodic in k, then all quantities at depth $(n+1)\Delta$ will also be periodic in k. This has two advantages: (1) the infinite linear convolution becomes a finite cyclic convolution; and (2) the discrete Fourier transform may be used to perform all Fourier transforms. Since periodicity in one Fourier domain is equivalent to discreteness in the other Fourier domain, the problem has effectively been discretized laterally as well as vertically: the quantites propagated in (2.11) are not samples of a bandlimited function, but actual discrete values. As $\Delta_k \to 0$, the situation approaches the continuous problem.

2.4 Born Approximation to the Layer Stripping Algorithm

It is worth noting how the Born approximation applies to the layer stripping equations (2.7). The Born approximation is a linearization of the inverse potential problem; the idea is to render the potential to be linearly related to the scattering data. This has been discussed in detail elsewhere; here we merely scale the potential by a small parameter ϵ, expand $\tilde{p}(x,z,t)$, etc. in a Taylor series in ϵ, and discard all terms of order ϵ^2 or smaller. The result is elimination of the product in (2.7b); since this is the one nonlinearity in (2.7) its elimination is not surprising. Combining the modified (2.7a) and (2.7b) and keeping (2.7c) results in

$$\left(\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial z^2}-\frac{\partial^2}{\partial t^2}\right)\tilde{q}(x,z,t)=0 \tag{2.14a}$$

$$V(x,z)=-2\tilde{q}(x,z,t=z^+). \tag{2.14b}$$

We recognize (2.14a) as the migration operator relating the wave field at the surface $z=0$ to the wave field on the plane parallel to the surface at depth z, and (2.14b) as the imaging operator (gradient) applied to the migrated wave field. Taking two Fourier transforms of (2.14a) with respect to t and x and using (2.4) and (2.9) yields

$$\left(\frac{\partial^2}{\partial z^2}+(k^2-k_x^2)\right)\bar{q}(k_x,z,k)=0; \quad \bar{q}(k_x,z,k)=\mathcal{F}_{t\to k}\mathcal{F}_{x\to k_x}\{\tilde{q}(x,z,t)\} \tag{2.15}$$

a differential equation which has the solution

$$\bar{q}(k_x,z,k)=\bar{q}(k_x,0,k)e^{-i\sqrt{k^2-k_x^2}z}. \tag{2.16}$$

The operation of the Born approximation to the layer stripping equations is now clear: (1) migrate the wave field from the surface to depth z; and (2) image the wave field at depth z to obtain the scattering potential. Note that imaging the potential requires taking the gradient of the wave field; this is why q, not p, is used. Note also that multiple scattering, which is inherently nonlinear, is neglected in (2.14) and (2.16). The coupling induced by the product term in (2.7) accounts precisely for all multiple scattering. More details about the Born approximation and its relation to layer stripping and integral equation methods is available in [4] and [7].

From the Schrodinger equation (2.1), it is apparent that for large wavenumbers k the potential $V(x,z)$ will be relatively small, and that multiple scattering will be less significant. Indeed, in the limit $k \to \infty$ the Born approximation becomes exact, in that multiple scattering effects become negligible. However, inversion based solely on asymptotically large k is clearly unstable; "exact" inverse scattering methods use low-wavenumber data as well as high-wavenumber data to stabilize the reconstruction. Also, it is clear that multiple scattering is more significant for small k ($V(x,z)$ is relatively large), so lack of high-wavenumber data makes the use of "exact" methods even more imperative.

3. Forward Problem Algorithm

3.1 Invariant Imbedding Algorithm

The invariant imbedding algorithm of [5] was used to generate the scattering data, to be input into the layer stripping algorithm. We briefly review this algorithm here, following the notation of [5] for convenience. Let k be wavenumber, as in the Schrodinger equation (2.1), $q = k_x$ (lateral wavenumber), $k(q) = \sqrt{k^2 - q^2}$ (vertical wavenumber, as in (2.16)), and p be lateral wavenumber of the incident plane wave (ultimately we are interested in $p = 0$). Then further define $h(z,q) = \hat{V}(z,k_x)$ (scattering potential) and $u(z,q) = \bar{p}(k_x,z,k)$ (wave field; see (2.15)). A slight problem with the notation of [5] is that the dependence of $u(z,q)$ and $R(c,q,p)$ on k is not explicit.

Finally, define $R(c,q,p)$ as the near-field planar reflection response, in direction q, of the portion of the medium below depth c, to an impulse $\delta(q-p)e^{-ik(q)z}/k(p)$, in direction

p (recall directions are specified by wavenumbers). Two *inverse* Fourier transforms taking $k \to t$ and $q = k_x \to x$, as in (2.15), transform $\delta(q-p)e^{-ik(q)z}$ into the impulsive plane wave $\delta(t - z\cos\theta - x\sin\theta)$, where θ is the angle of incidence (measured from the vertical) defined from $p = k\sin\theta$. Hence $k(p)R(0,q,0)$, computed for each k and then inverse Fourier transformed as in (2.4), is precisely the reflection response to an impulsive plane wave normally incident on the medium.

We sketch through the derivation of the invariant imbedding equations to show the similarities and differences to layer stripping. A Fourier transform of the Schrodinger equation (2.1) taking $x \to q = k_x$ yields (recall $h(z,q) = \hat{V}(z,k_x)$)

$$\left(\frac{\partial^2}{\partial z^2} + k^2(q) - h(z,q)*\right) u(z,q) = 0, \tag{3.1}$$

where $*$ denotes convolution in q and $k^2(q) = k^2 - q^2$. Defining

$$v(z,q) = \left(u(z,q) + \frac{\partial u}{\partial z}\frac{1}{ik(q)}\right)/2; \quad w(z,q) = \left(u(z,q) - \frac{\partial u}{\partial z}\frac{1}{ik(q)}\right)/2, \tag{3.2}$$

it can be shown ([5], p.93) that $v(z,q)$ and $w(z,q)$ satisfy the coupled system (compare to (2.8))

$$\frac{d}{dz}\begin{bmatrix} v \\ w \end{bmatrix} = \begin{bmatrix} ik(q)v - (h*(v+w))/(2ik(q)) \\ -ik(q)w + (h*(v+w))/(2ik(q)) \end{bmatrix} \tag{3.3}$$

where all variables are functions of (z,q).

Now *imbed* the system (3.3) as follows. Let $A(z,c,q,p)$ and $B(z,c,q,p)$ satisfy (3.3), initialized with $A(z=c,c,q,p) = \delta(q-p)/k(p)$ and $B(z=L,c,q,p) = 0$ (the latter is a radiation condition; recall $V(x,z)$ has support in $0 < z < L$). Then

$$v(z,q) = kA(z,0,q,p); w(z,q) = kB(z,0,q,p);$$

$$A(c,c,q,p) = \delta(q-p)/k(p); R(c,q,p) = B(c,c,q,p) \tag{3.4}$$

Furthermore, $\frac{\partial A}{\partial c}$ and $\frac{\partial B}{\partial c}$ also satisfy (3.3), but with initial conditions

$$\frac{\partial A}{\partial c}(c,c,q,p) = -\frac{\partial A}{\partial z}(c,c,q,p); \quad \frac{\partial B}{\partial c}(L,c,q,p) = 0. \tag{3.5}$$

By superposition, the solution to (3.3) with these initial conditions is

$$\frac{\partial A}{\partial c}(z,c,q,p) = -\int A(z,c,q,q')k(q')\frac{\partial A}{\partial z}(c,c,q',p)dq' \tag{3.6a}$$

$$\frac{\partial B}{\partial c}(z,c,q,p) = -\int B(z,c,q,q')k(q')\frac{\partial A}{\partial z}(c,c,q',p)dq'. \qquad (3.6b)$$

([5],p.95). We also have from the last of (3.4)

$$\frac{dR}{dc}(c,q,p) = \frac{dB}{dc}(c,c,q,p) = \frac{\partial B}{\partial z}(c,c,q,p) + \frac{\partial B}{\partial c}(c,c,q,p). \qquad (3.7)$$

Finally, setting $z = c$ in (3.3), substituting into (3.6), and substituting again into (3.7) gives the following *invariant imbedding equation for R(c,q,p)*:

$$i\frac{dR}{dc}(c,q,p) = (k+k(q))R(c,q,p)+h(c,q-p)/(2k(q)k)+\int\int R(c,q,q')h(c,q'-q'')R(c,q'',p)/2\,dq'\,dq''$$

$$+\int (h(c,q-q')R(c,q',p)/k(q) + R(c,q,q')h(c,q'-p)/k)/2\,dq'; \qquad R(L,q,p) = 0. \quad (3.8)$$

This is (11a) on p. 97 of [5].

Note that (3.8) is computed recursively in decreasing c, starting at $c = L$ and ending at $c = 0$. This must be done for each p, q, and k (recall that $R(c,q,p)$ also depends on k; this dependence is not shown explicitly in (3.8) since none of the integrations are over k). Having computed $R(0,q,p)$ for all k, i.e., having computed $R(0,q,p,k)$, the inverse Fourier transform (2.4) of $kR(0,q,0,k)$ is precisely the reflection response to an impulsive plane wave normally incident on the medium. This is the scattering data used as input to the layer stripping algorithm.

3.2 Numerical Implementation of Invariant Imbedding Algorithm

Despite its apparent complexity, (3.8) can be implemented numerically in a straightforward manner by discretization similar to that used to obtain (2.11) from (2.8). Since (3.8) is already in the wavenumber domain, and the scattering potential $h(z,q)$ is known exactly, no computational instability issues arise. The integrals may be evaluated using the trapezoidal rule, and a backward difference approximation to $\frac{dR}{dc}$ used to propagate (3.8) in *decreasing* c from $c = L$ to $c = 0$.

Once again we assume a periodicity of $1/\Delta$ in the values of all functions of wavenumbers; this corresponds to the discretized functions being actual discrete values, rather than sampled values of bandlimited continuous functions. The infinite integrals in (3.6)

and (3.8) become cyclic integrals (computed only over one period), so their evaluation is straightforward. The multiplication by $k + k(q)$ in (3.8) is windowed to zero for values greater than $1/(2\Delta)$, as in (2.10), and then periodically extended.

Note that it is not possible to compute the reflection response for $k = 0$ or $k(q) = 0$, due to the divisions by these in (3.8). The former can be assumed to be zero, since a non-zero DC reflection response would represent *permanent* displacement resulting from the impulsive plane wave! The latter corresponds to incidence at 90 degrees, which would not create a backscattered field in the $+z$ direction. Hence omitting these does not present a problem.

3.3 Born Approximation to Invariant Imbedding Algorithm

The invariant imbedding equation (3.8) is suggestive of a 2-D version of the Riccati equation familiar in 1-D scattering in layered media. The two integral terms correspond to the square term in the 1-D Riccati equation. To aid in understanding (3.8), we now apply the Born approximation to (3.8), and show that the Born approximation to the layer stripping algorithm (2.14b) and (2.16) reconstructs the potential from the reflection response generated by the Born approximation to the invariant imbedding equation (3.8).

As in Section 2.4, we scale the potential by a small parameter ϵ, expand the wave field and reflection response in a Taylor series in ϵ, and discard all terms of order ϵ^2 or smaller. The result is elimination of the two integrals of products terms in (3.8), leaving

$$i\frac{dR}{dc}(c,q,p) = (k + k(q))R(c,q,p) + h(c,q-p)/(2k(q)k); \qquad R(L,q,p) = 0. \tag{3.9}$$

Since there is no longer coupling between $R(c,q,p)$ of different p, we can set $p = 0$ (normal incidence) and solve the differential equation (3.9), yielding

$$kR(z,q,k) = -ie^{-i(k(q)+k)z}\int_L^z \frac{\hat{V}(z',q)}{2k(q)} e^{i(k(q)+k)z'}dz'. \tag{3.10}$$

The factor of k multiplying $R(z,q,k)$ is present because $k(p)R(z,q,p,k)$ is the Fourier transform of the reflection response to an impulse, as discussed in the second paragraph of Section 3.1. Since $p = 0$ here, we have $k(0) = k$, so $kR(z,q,k)$ is the frequency-domain reflection response to a planar impulse.

Equation (3.10) has a very clear interpretation: to form the reflection response at depth z in the Born approximation, assume the incident impulsive plane wave penetrates without being scattered to each depth z', and is then scattered by the potential $\hat{V}(z',q)$ at that depth. Then use the migration operator $e^{ik(q)z}$ to migrate each scattered field back to depth z independently (neglecting all coupling), and superpose the scattered fields due to each $\hat{V}(z',q)$. At the surface $z=0$ this is clear, but it applies to any depth z.

Note that $\bar{q}(k_x,z,k)$ in (2.16) in the time domain is causal for all z (see Fig. 2b) while a time delay/advance $e^{-ikz'}$ must be included in (3.10). Also note that $\hat{V}(z',q)$ in (3.10) is scaled by $-i/(2k(q))$; the reason for this will become apparent in (3.13) below.

Now consider the Born approximation to the layer stripping algorithm (2.14b) and (2.16) applied to (3.10). Taking the Fourier transform (2.4) of (2.8a) and using (2.16) gives

$$\bar{q}(z,q,k)=\left(\frac{\partial}{\partial z}+ik\right)\bar{p}(z,q,k)=\left(\frac{\partial}{\partial z}+ik\right)\bar{R}(z,q,k)e^{ik(q)z}. \tag{3.11}$$

Inserting (3.10) into (3.11) shows that the Born-approximated layer stripping algorithm computes

$$\bar{q}(z,q,k)=-i/(2k(q))\hat{V}(z,q)e^{ik(q)z} \tag{3.12}$$

from the Born-approximated scattering data. Using (2.14b) shows that the Born-approximated layer stripping algorithm computes

$$-2\mathcal{F}^{-1}_{k\to t}\{\bar{q}(z,q,k)\}_{t=z}=\hat{V}(z,q)\mathcal{F}^{-1}_{k\to t}\{\frac{e^{ik(q)z}}{-ik(q)}\}_{t=z}=\hat{V}(z,q) \tag{3.13}$$

so that it does indeed correctly compute the scattering potential $\hat{V}(z,q)$ in the Born approximation.

4. Numerical Results

4.1 Initialization

The algorithm described in Section 3 was used to generate the backscattered reflection response $\bar{R}(0,q,k)$ to an impulsive plane wave for several different scattering potentials

$V(x,z)$. The inverse Fourier transform (2.4) $\frac{\partial}{\partial t}\check{R}(0,q,t)$ of $kR(0,q,k)$ was then used to initialize the discrete layer stripping algorithm of Section 2, with (recall $q = k_x$)

$$p(0,t,k_x) = \check{R}(0,q=k_x,t); \quad q(0,t,k_x) = 2\frac{\partial}{\partial t}\check{R}(0,q=k_x,t). \tag{4.1}$$

The latter initial condition comes from (2.7a) and the fact that $\check{R}(z,q,t) = \check{R}(0,q,t+z)$ in the homogeneous overlying half-space $z < 0$, since $R(0,q,t)$ is a *backscattered* (i.e., upward-traveling) wave. Note that the sample applications of Section 2 would require different initial conditions.

4.2 Forward Problem vs. Inverse Problem Algorithms

The invariant imbedding algorithm was used to generate the forward data so that the layer stripping inverse problem algorithm would not simply run the computations of the forward problem algorithm backwards. Although the two algorithms must of course be mathematically equivalent, since they are both "exact," they are derived from different mathematical principles.

Some specific differences between the forward problem (invariant imbedding) algorithm (FPA) and the inverse problem (layer stripping) algorithm (IPA) are as follows:

1. The FPA propagates the reflection coefficient at depth $R(c,q,p,k)$. The IPA propagates the field and field gradient $p(z,t,k_x)$ and $q(z,t,k_x)$. Note that $R \neq q/p$, since R is the ratio of downgoing and upgoing ***waves***, not field quantities;
2. The FPA operates in the k (frequency) domain, while the IPA operates in the t (time) domain;
3. The FPA computes $R(c,q,p,k)$ for all c,q,p,k, while the IPA is initialized using $kR(0,q,0,k)$, a slice of the FPA function. Note in the FPA (3.8) the integrals over q' and the differences $q'-q''$; these clearly have no counterpart in the IPA;
4. The FPA propagates (3.8), which can be viewed as a 2-D generalization of the Riccati equation familiar in 1-D inverse scattering. The IPA propagates the coupled system (2.8); note that this differs from the coupled system (3.3) used to derive (3.8);
5. While both algorithms are discretized in depth, the FBP results did not vary significantly with mesh size, so this should not be an issue. The IBP results also did not

vary significantly with mesh size, and gave good results at several different resolutions (see below);

These differences make it clear that errors are not cancelling out algebraically between the FPA and the IPA, i.e., the IPA is not effectively running the FPA computations backwards.

4.3 Summary of Results

The numerical performance of the layer stripping algorithm was studied under a variety of conditions. The results may be summarized as follows:

1. The layer stripping algorithm successfully reconstructed the potential in the absence of noise. The only difficulty was due to the smoothing of the transverse derivative, which slightly smoothed very sharp variations in the lateral direction;
2. The layer stripping algorithm continued to work well when a small amount of Gaussian random noise was added to the reflection response. The reconstructed potential was slightly degraded, of course, but the amount of degradation seemed to vary smoothly with the amount of noise–a slight increase in noise level did not vastly degrade the reconstructed potential;
3. The layer stripping algorithm reconstructions were superior to those using the Born approximation (as specified in Section 2.4 above), in that the Born approximation treated multiple scattering events as additional single scattering events, resulting in errors in the reconstructed potential, particularly for large z. This effect was more pronounced when the potential had numerically large values; for small potentials $V(n\Delta, k\Delta)\Delta << 1$ the Born approximation worked quite well. This was as expected; multiple scattering involves products of potentials, and multiplying small values results in even smaller values;
4. The performance of the algorithm seemed to vary little with the size of the discretization length Δ, provided that the same Δ was used in the discretized invariant imbedding algorithm. This suggests there may be a close relation between the discretized versions of these algorithms. Coarse grid reconstructions seemed to be merely under-

sampled versions of the fine grid reconstructions; the basic features of the reconstructions were identical.

We illustrate these points with some numerical examples below. It should be noted that the following is only a representative and illustrative sample of our results; the above conclusions are not based merely on the results below. Unless otherwise specified, all examples used $\Delta = 1/32$, $L = L_x = 1/2$, and $\Delta_k = 1$. The 3-D plots are depicting 2-D functions $V(x, z)$; they do not represent objects buried in a homogeneous surrounding medium.

4.4 Comparison with Born Approximation

The potential $V(x, z)$ is shown in Fig. 3a. Note that this is a smooth, rounded potential having compact support in both x and z.

The reconstructed potential using the Born approximation is shown in Fig. 3b. Although Fig. 3b superficially seems to be identical to Fig. 3a, study carefully the deepest part of the reconstructed $V(x, z)$. The original $V(x, z)$ is zero for $z > 24/32$, while the Born-reconstructed $V(x, z)$ does not become zero until $z > 26/32$; it has a "tail." This "tail" is caused by multiple scattering that is interpreted under the Born approximation as primary scattering due to an additional non-zero portion of the scattering potential; actually, there is no such portion.

The reconstructed potential using the layer stripping algorithm is shown in Fig. 3c. This reconstruction has no "tail"; the multiple scattering that produces it has been accounted for in the algorithm and eliminated. The reconstruction is almost perfect.

A different potential is shown in Fig. 4a. Note that this potential function is constant over a central "plateau," and then drops off rapidly to zero.

The reconstructed potential using the Born approximation is shown in Fig. 4b. Note again the presence of a "tail" at its deepest part, while there is no "tail" at its shallowest part, since multiple scattering has not yet had *time* to occur in this part of the time-domain impulse response (the lack of symmetry in z is apparent if one looks at the figure as a whole). Also note the problems in reconstructing the lateral edges of the potential

function; the central "plateau" is much smaller than it should be.

The reconstructed potential using the layer stripping algorithm is shown in Fig. 4c. Again the "tail" caused by multiple scattering has been eliminated. However, the shallowest and deepest edges of the "plateau" have been rounded off slightly. Since this is symmetric between the shallowest and deepest parts, it is not due to multiple scattering. We attribute it to smoothing in the transverse derivative.

4.5 Effects of Additive Noise

The potential $V(x,z)$ used in Figs. 3 was scaled as shown in Fig. 5a, and Gaussian random noise was added to the reflection response $R(0,q,k)$. The signal-to-noise ratio, computed as the square root of the sum of the squares of the discrete signal values divided by the square root of the sum of the squares of the discrete noise values, was found to be 36 dB for one run and 18 dB for another (to get power SNR these values should be doubled). Note that any powers of Δ and numbers of points being averaged will cancel in this ratio.

The reconstructions at 36 dB are virtually perfect; in fact, the reconstructions shown in Figs. 3 are actually these reconstructions. The reconstructions at 18 dB are shown in Fig. 5b using the Born approximation and Fig. 5c using the layer stripping algorithm. Note that even in these noisy reconstructions the "tail" is still a significant feature in the Born approximation reconstruction, while the layer stripping reconstruction has correctly removed the "tail."

To see the degradation of the layer stripping algorithm in the presence of increasing amounts of noise added to the reflection response, study Figs. 6. Fig. 6a shows the original potential function, which is the same as Fig. 4a. Fig. 6b shows a noisy reconstruction of the potential function shown in Fig. 6a, and in Fig. 6c the signal-to-noise ratio has been reduced by a factor of four. The increasing degradation of the reconstruction is obvious, but the layer stripping algorithm does not fall apart even in large amounts of additive noise.

A similar study is carried out for a different potential function in Figs. 7. Fig. 7a shows

the original potential function, and Figs. 7b and 7c correspond to Figs. 6b and 6c. The only notable feature of the layer stripping reconstructions is the slight (one pixel wide) "shelf" induced by the smoothed transverse derivative (this is discussed in more detail below); otherwise, the reconstructed potential smoothly degrades in increasing noise.

Note the presence of the sharp ridge along the line $z = 0$ in Figs. 6 and 7. This ridge is due to the non-zero mean of the noise being added to $R(0, q, k)$. When $V(x, z)$ is computed by taking the inverse Fourier transform (2.9), this non-zero mean, a constant in the Fourier wavenumber k domain, becomes an impulse in the spatial z domain. This impulse is the ridge.

4.6 Discussion of Numerical Stability in Noise

The smooth degradation of the reconstructed potential with increasing noise levels might seem surprising, since the inverse scattering problem is known to be ill-conditioned. The reason for this is that multiple scattering has a relatively small (compared to single scattering) effect, so that the Born approximation result will be approximately the same as the layer stripping result. The Born approximation is linear, so that any noise added to the reflection response will produce an addition to the reconstructed potential whose strength is directly proportional to the noise strength (halving the noise will halve the addition); hence the Born-reconstructed potential will degrade smoothly, and it is not surprising that the reconstructed potential from layer stripping also degrades smoothly.

This heuristic argument should not be taken too far; in the 1-D case, it is well known that large noise levels can cause severe problems in layer stripping algorithms, and indeed in *any* "exact" method. The reason for this is NOT numerical instability, as is commonly believed; the 1-D layer stripping algorithm is identical to the Schur algorithm (see [8]), which is ***known to be numerically stable.***

The reason that 1-D layer stripping algorithms can give unstable results when they are applied to noisy reflection data is as follows. It is well known that the free-surface reflection response of a 1-D layered medium to an impulsive plane wave below the surface is one side of the autocorrelation of its transmission response; hence it must be positive

semi-definite. Noise added to the reflection response can make the two-sided response (the reflection response added to its time reversal) become non-positive-semi-definite, in which case it is no longer the reflection response to *any* layered medium. The problem is now ill-posed, in the sense of having no solution; it is not surprising that the layer stripping algorithms become unstable.

However, small amounts of additive noise will not cause the reflection response to become non-positive-semi-definite; as long as this is true, the layer stripping algorithms will behave well numerically. Our results in this paper suggest that a a similar situation is present in the 2-D inverse scattering problem considered here; this is a topic of current research.

4.7 Smoothed Reconstructions Due to Smoothed Transverse Derivative

The smoothing in the transverse derivative incurred by using the clipped filter (2.10) causes a slight but noticeable smoothing of $V(x,z)$ along the x direction. This was manifested in the reconstructions in Figs. 7 by the "shelf" that appeared at the ends of the reconstructed potential function. Another example of this is illustrated in Figs. 8. Fig. 8a shows the original potential function, which was produced by taking the potential function of Fig. 7a and adding random noise to it. The idea here is that in real life potential functions will not have simple analytic forms; they will be complicated functions. Hence Fig. 8a is closer to a realistic potential function.

The reconstructed potential from the layer stripping algorithm is shown in Fig. 8b. Note again the one-pixel-wide "shelf" at each of the two flat ends of the potential function. We attribute this to the smoothing of the transverse derivative in the layer stripping algorithm; unable to reconstruct the sharp jump from zero, the algorithm provides a laterally smoothed reconstruction in which the reconstructed potential takes two smaller lateral jumps instead of a single large jump. Note that the "shelf" is half the height of the jump in x at each depth z.

Also note in Fig. 8b that the "noisy" part of the potential in Fig. 8a has been noticeably smoothed. This again seems to be due to the smoothing in the transverse derivative;

note that the reconstructed potential is "rougher" in the z direction (for which there is no smoothing) than in the x direction (in which there is smoothing). This smoothing effect should be taken into consideration in potentials reconstructed using layer stripping algorithms.

4.8 Effect of Discretization Length Δ

The above numerical runs all used $\Delta = 1/32$. Results for a larger $\Delta = 1/16$ are shown in Figs. 9. Fig. 9a shows the original potential, which is an undersampled version of the potential in Fig. 4a. The reconstructed potential using the Born approximation is shown in Fig. 9b. Note how poorly the Born approximation reconstructs the central "plateau" of the potential function. The reconstructed potential using the layer stripping algorithm is shown in Fig. 9c. Although the central "plateau" is reconstructed quite well, the potential function as a whole is spread out one pixel in each direction. This shows that while the invariant imbedding and layer stripping algorithms are clearly closely connected, the discretized layer stripping algorithm is NOT merely running the invariant imbedding algorithm backwards.

Results for a smaller $\Delta = 1/64$ are shown in Figs. 10. Fig. 10a shows the original potential, which is a more finely sampled version of the potential in Fig. 4a. The reconstructed potential using the layer stripping algorithm is shown in Fig. 10b. The reconstruction is almost perfect–even the lateral smoothing caused by the smoothed transverse derivative is not apparent. This is due to the fact that although $\Delta_k = 1$, the maximum value of k_x is now 32 instead of 16; the smoothing starts at a much higher wavenumber. A very close comparison of Figs. 10a and 10b show that the reconstruction is not quite perfect; the reconstructed potential is still spread out one pixel in each direction. But this effect is virtually negligible on this scale.

One final example combines a smaller Δ, additive noise in the reflection response, and smoothed reconstruction. Fig. 11a shows the original potential, which is a more finely sampled version of the potential in Fig. 7a. Random noise was added to the reflection data, at a signal-to-noise ratio of 15.7 dB. The reconstructed potential using the layer stripping

algorithm is shown in Fig. 11b. All the features discussed in Section 4.3 are again present in Fig. 11b. These include the "shelf," still one pixel wide, the ridge along the line $z = 0$, and the main shape of the potential function still visible in the noise. This shows that these effects occur at different discretization lengths, and indeed may be endemic to layer stripping reconstructions in noise for any Δ.

5. Conclusion

The numerical performance of the 2-D layer stripping algorithm of [4] has been studied for the first time. This represents the first numerical implementation of an "exact" non-iterative inverse scattering algorithm that includes the effects of *all* multiple scattering and diffraction effects. The forward scattering data were generated using the invariant imbedding algorithm of [5]. The results indicated that layer stripping is a viable technique for solving 2-D Schrodinger equation inverse potential problems, for which two applications were briefly reviewed.

Two particularly important results were that: (1) the "exact" reconstructions using the layer stripping algorithm are a noticeable improvement over the Born approximation reconstructions; and (2) small amounts of additive noise in the reflection response do not cause numerical instability in the layer stripping algorithm. The results were illustrated using several numerical examples. It was also shown for the first time that the Born approximation to the layer stripping algorithm reconstructs the scattering potential from the reflection response generated by the Born approximation to the invariant imbedding algorithm of [5].

Acknowledgment

It is a pleasure to acknowledge the help of Ms. Susan Wei, who helped with the preliminary stages of this research. This research was supported by the Office of Naval Research under grant # N00014-90-J-1897.

References

1. Coen S, Cheney M and Weglein A 1984 Velocity and density of a two dimensional acoustic medium from point source surface data *J. Math Phys.* **25** 1857-1860
2. Yagle A E 1987 A layer stripping fast algorithm for the two-dimensional direct current inverse resistivity problem *IEEE Trans. Geosci. and Rem. Sensing* **GE-25** 558-563
3. Cheney M 1984 Inverse scattering in dimension two *J. Math Phys.* **25** 94-102
4. Yagle A E and Levy B C 1986 Layer stripping solutions of multidimensional inverse scattering problems *J. Math Phys.* **27** 1701-1710
5. Wilcox R 1970 Wave propagation through longitudinally and transversally inhomogeneous slabs-I *Invariant Imbedding* ed by Bellman R E and Denman E D (New York: Springer)
6. Cheney M 1990 Stability analysis of the Yagle-Levy multidimensional inverse scattering algorithm *preprint* Dept. of Mathematical Sciences, Rensselaer Polytechnic Institute, Troy NY
7. Snieder R 1990 The role of the Born approximation in nonlinear inversion *Inverse Problems* **6** 247-266
8. Bruckstein A M, Levy B C, and Kailath T 1985 Differential Methods in Inverse Scattering *SIAM J. Appl. Math* **45** 312-335

Figure Headings

1. The 2-D inverse scattering problem

2a. Update pattern for $p(z, t, k_x)$

2b. Update pattern for $q(z, t, k_x)$

3a. Original potential function

3b. Born approximation reconstruction. Note the "tail" at the right.

3c. Layer stripping algorithm reconstruction

4a. Original potential function

4b. Born approximation reconstruction. Note the "tail" at the right.

4c. Layer stripping algorithm reconstruction

5a. Original potential function

5b. Noisy Born approximation reconstruction. Note the "tail" at the right.

5c. Noisy layer stripping algorithm reconstruction

6a. Original potential function

6b. Noisy layer stripping algorithm reconstruction

6c. Noisier layer stripping algorithm reconstruction

7a. Original potential function

7b. Noisy layer stripping algorithm reconstruction

7c. Noisier layer stripping algorithm reconstruction

8a. Original "noisy" potential function

8b. Layer stripping algorithm reconstruction

9a. Original potential function using $\Delta = 1/16$

9b. Born approximation reconstruction. Note poor reconstruction of the "plateau."

9c. Layer stripping algorithm reconstruction

10a. Original potential function using $\Delta = 1/64$

10b. Layer stripping algorithm reconstruction

11a. Original potential function using $\Delta = 1/64$

11b. Noisy layer stripping algorithm reconstruction

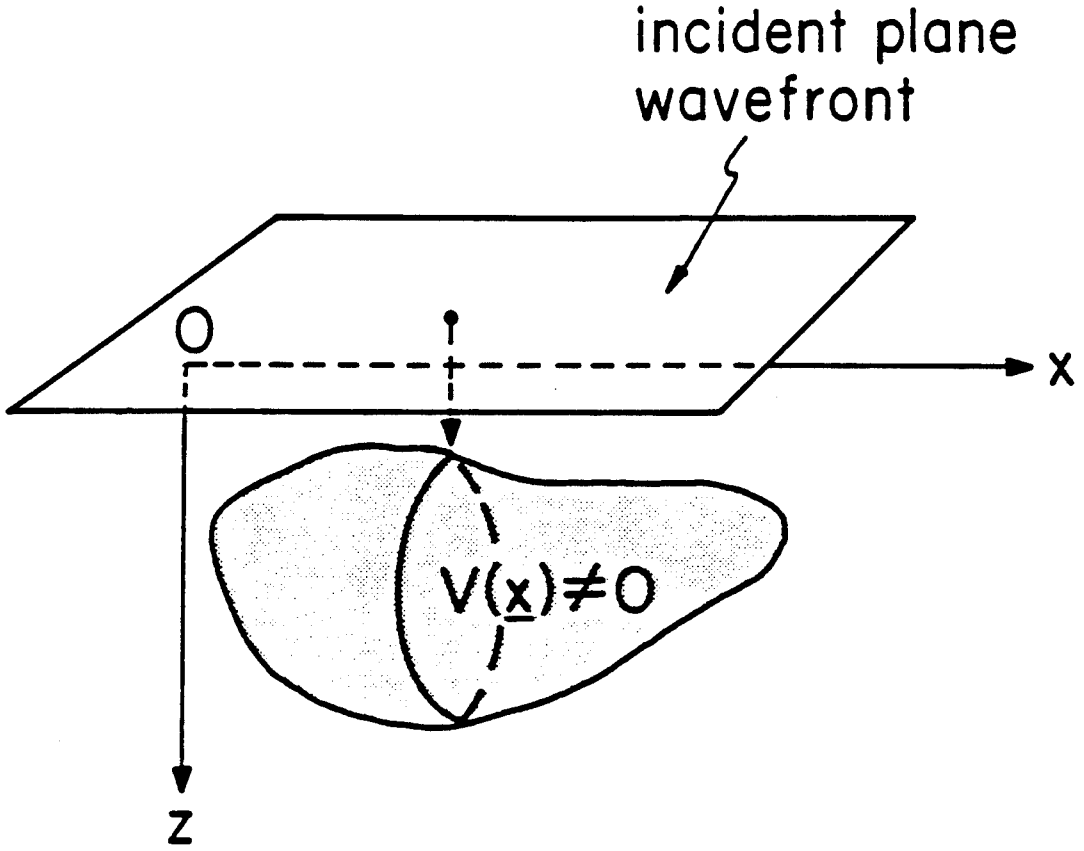


FIG. 1

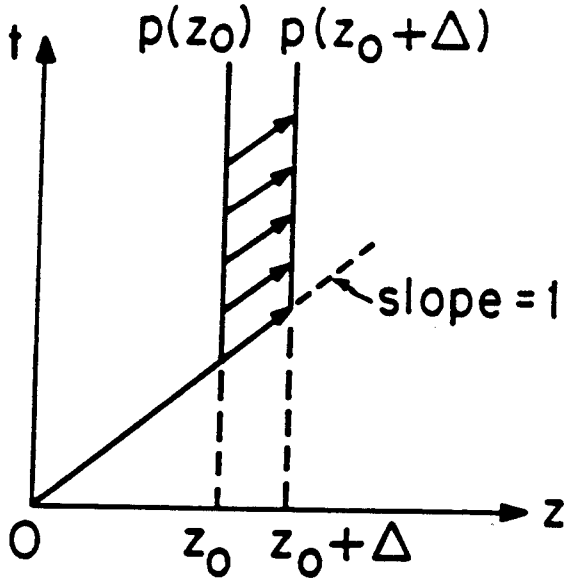


Fig. 2a

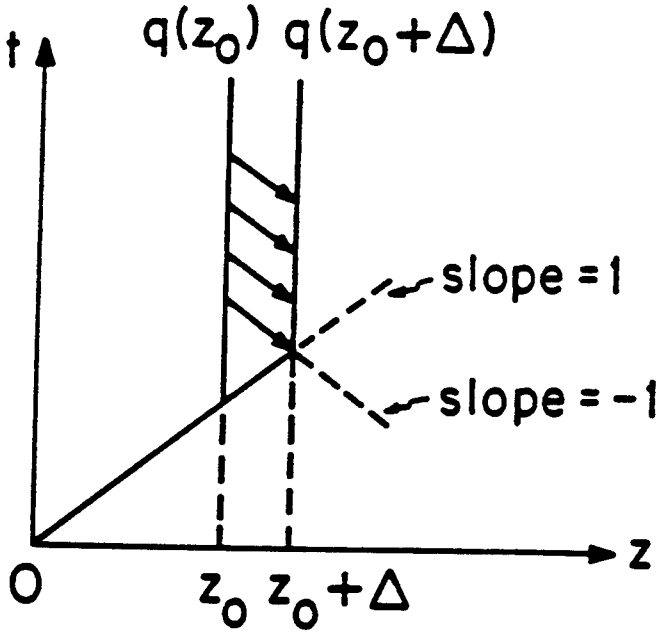


Fig. 2b

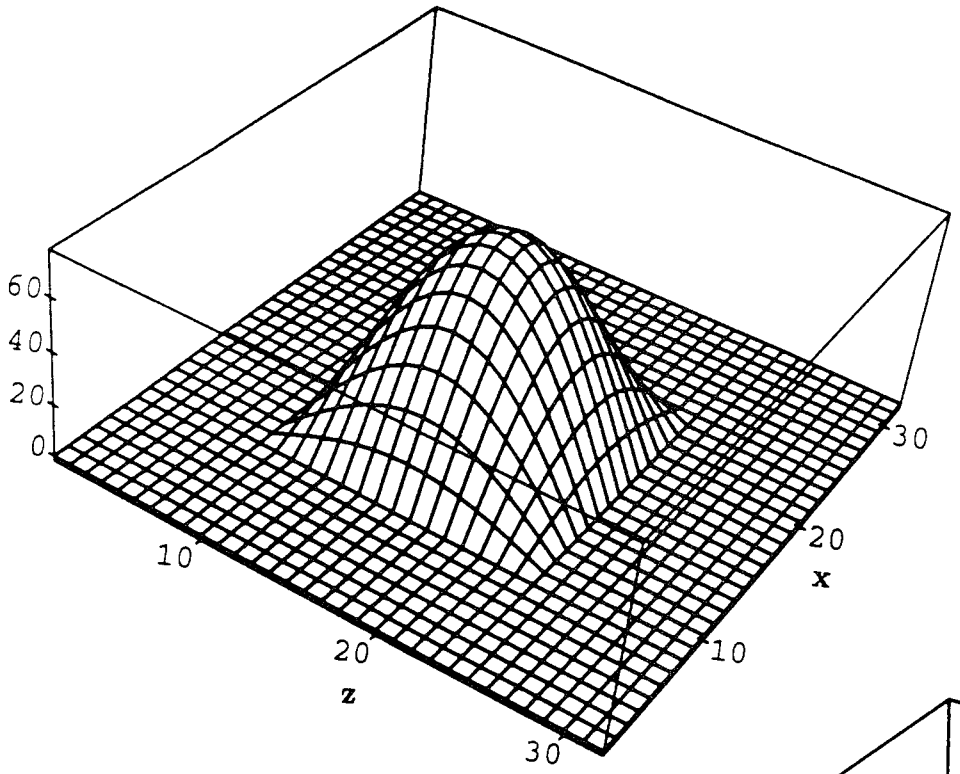


FIG. 3a

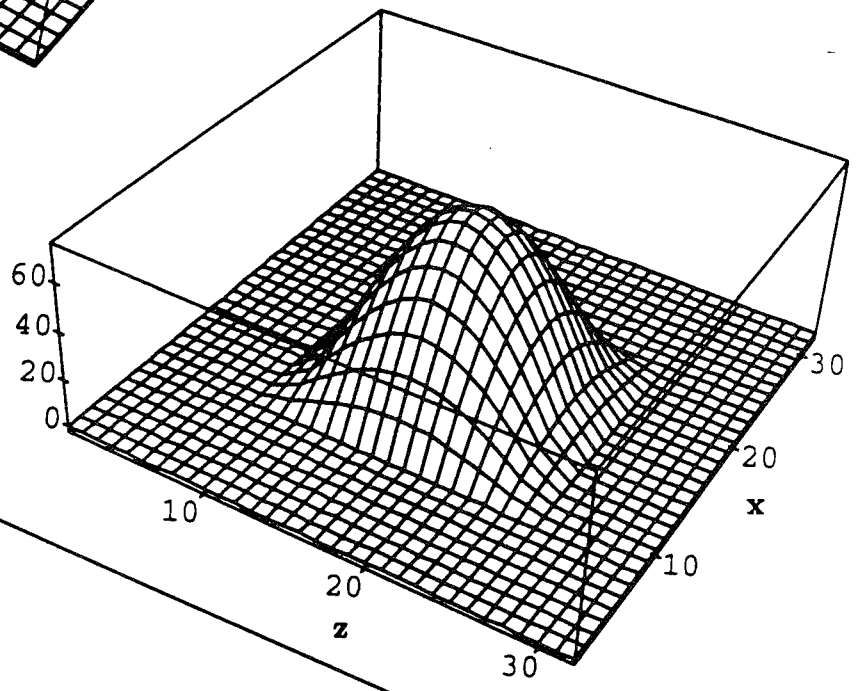


FIG. 3b

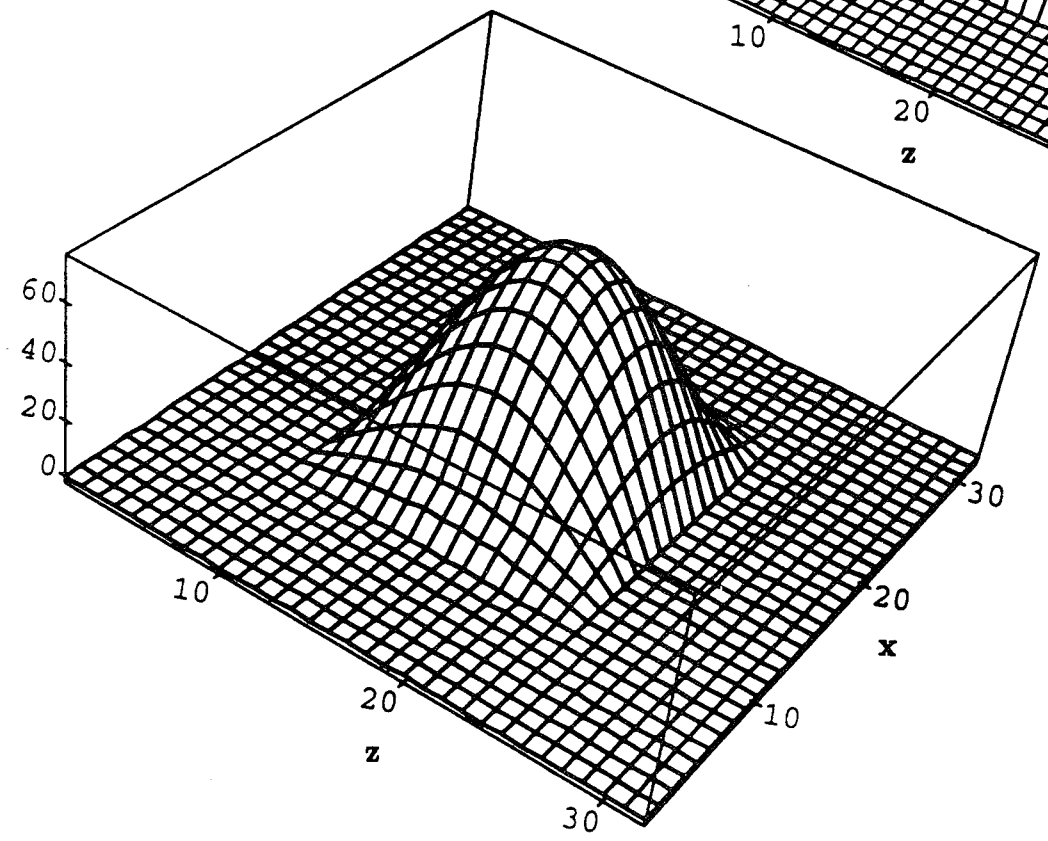


FIG. 3c

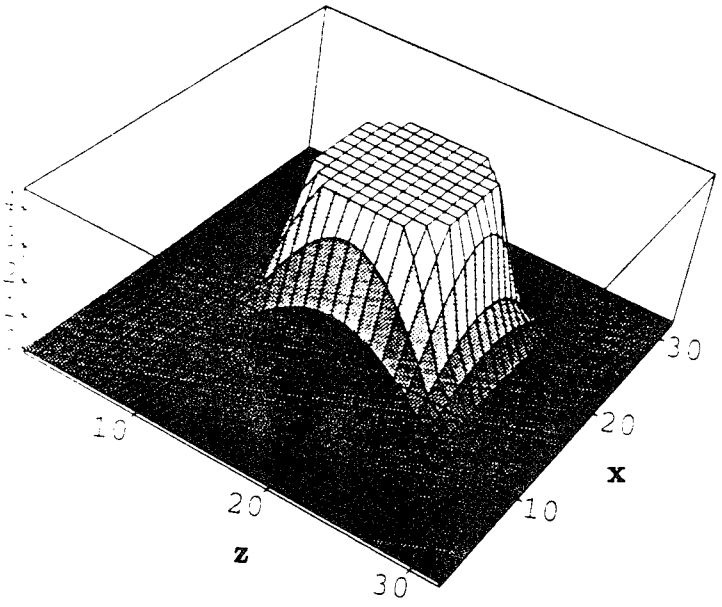


FIG. 4a

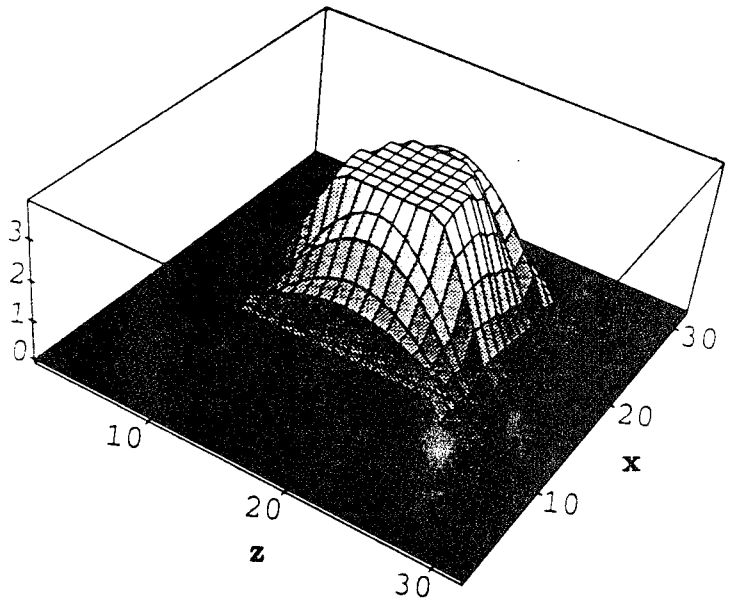


FIG. 4b

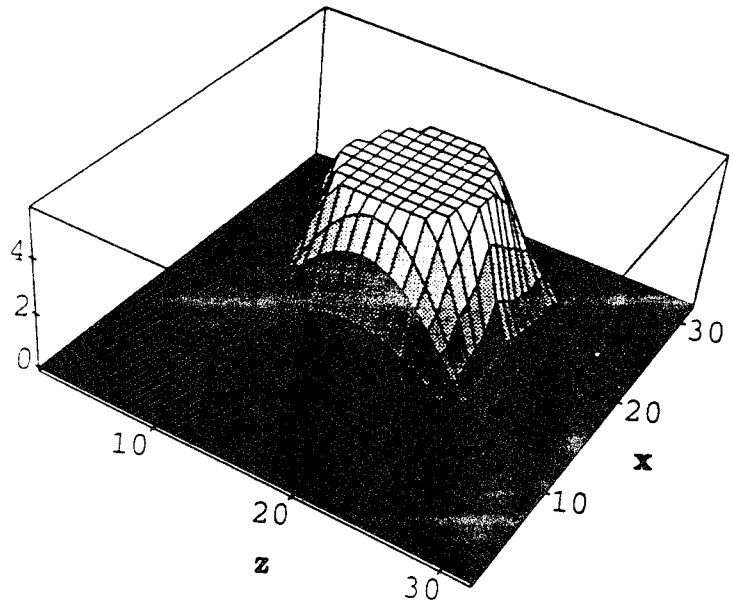


FIG. 4c

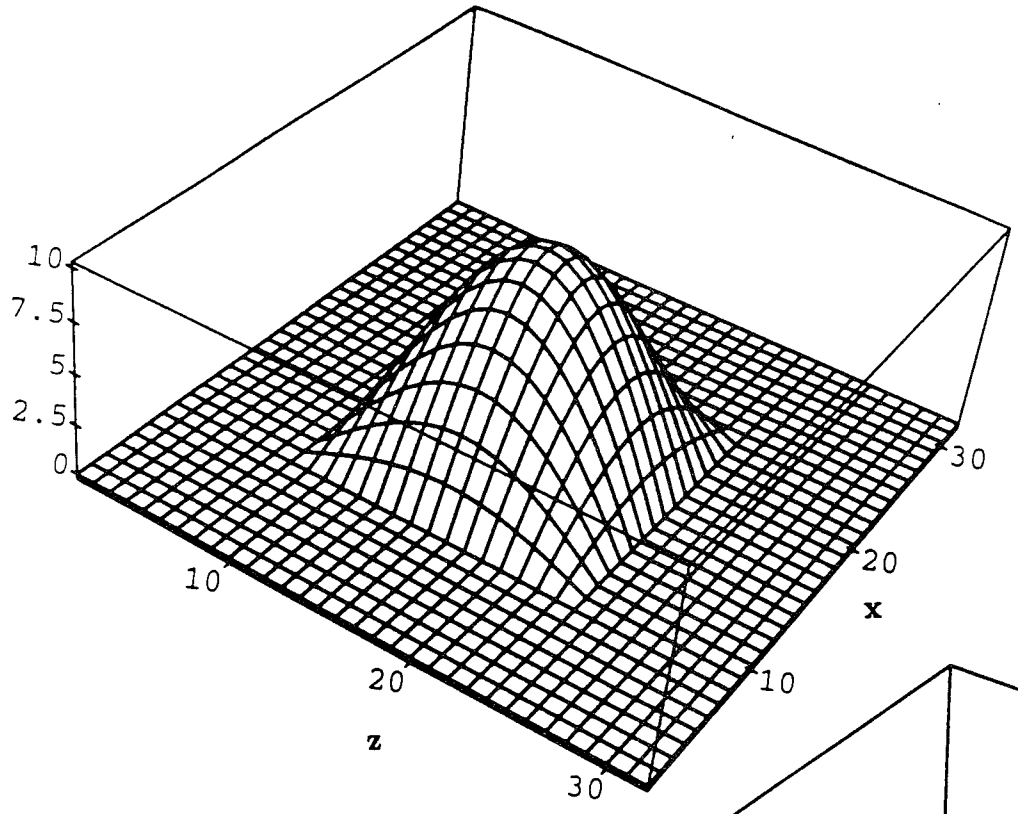


FIG. 5a

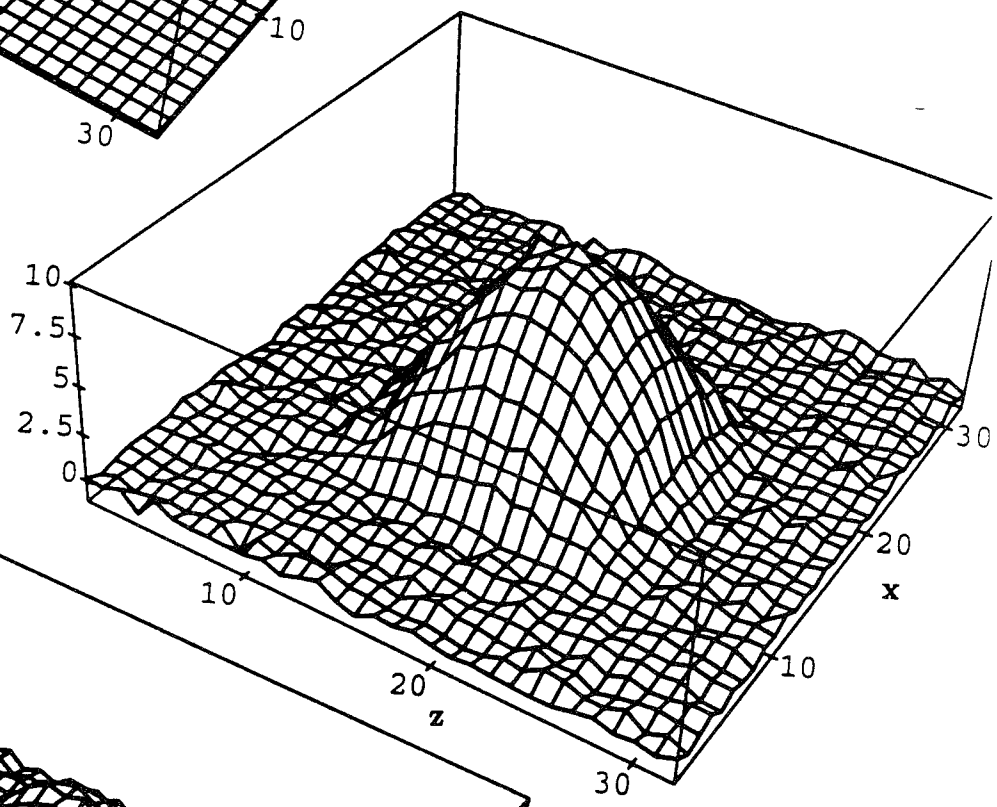


FIG. 5b

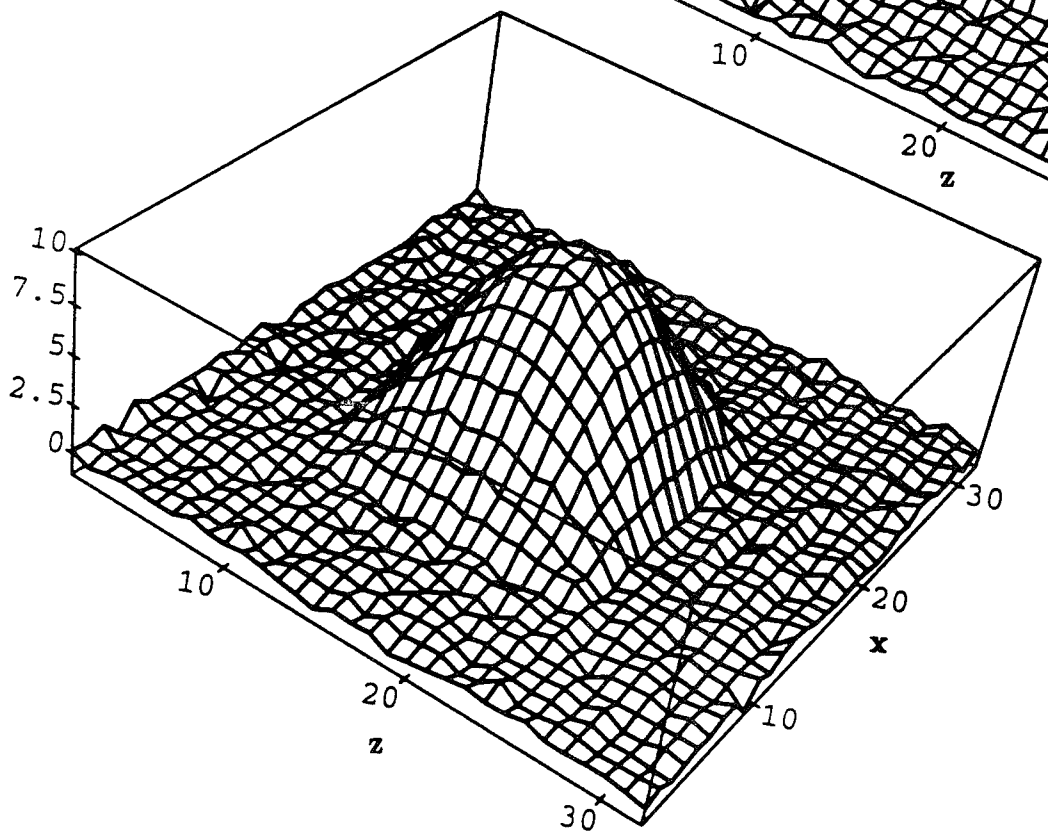


FIG. 5c

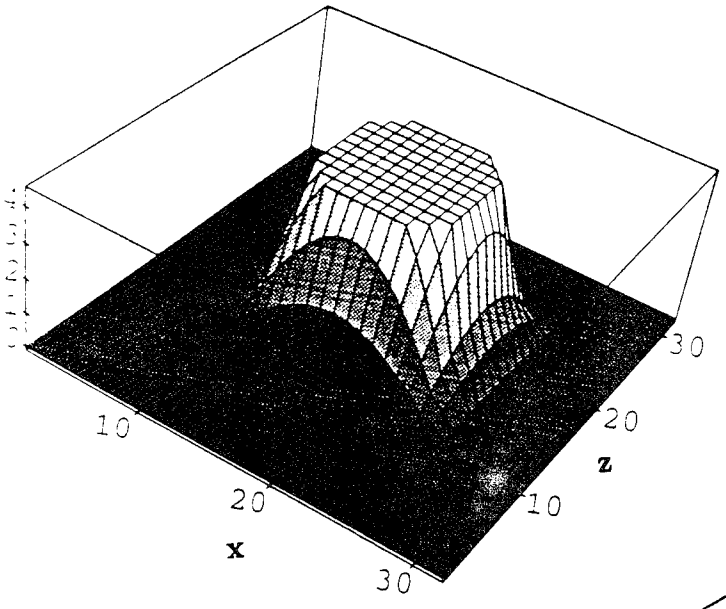


FIG. 6a

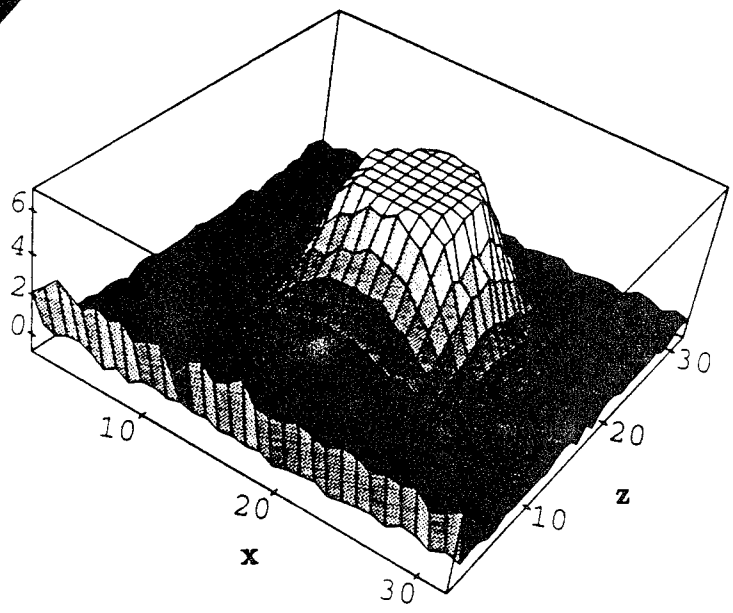


FIG. 6b

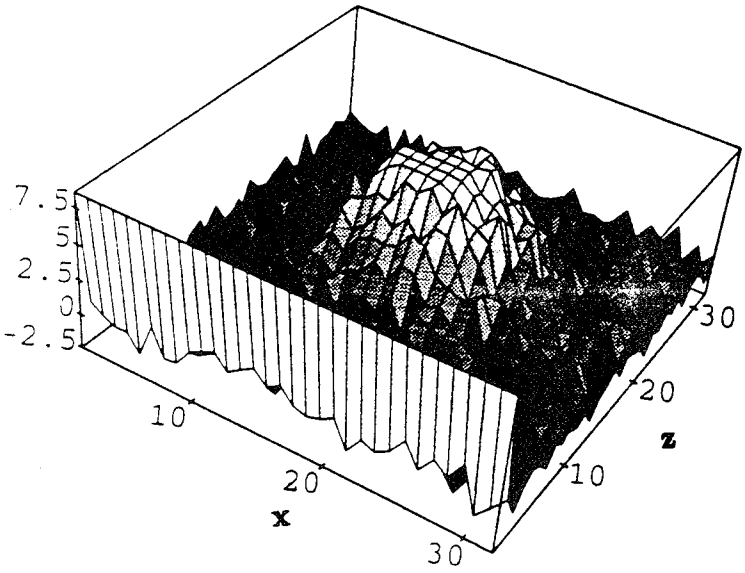


FIG. 6c

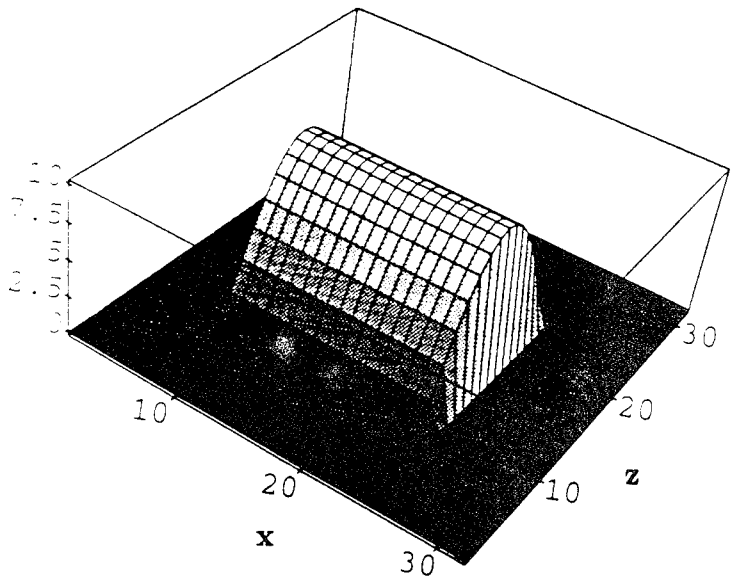


FIG. 7a

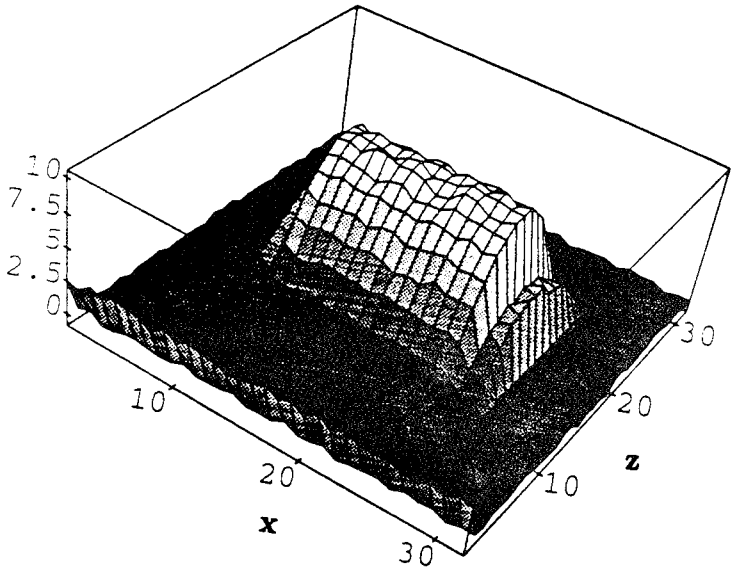


FIG. 7b

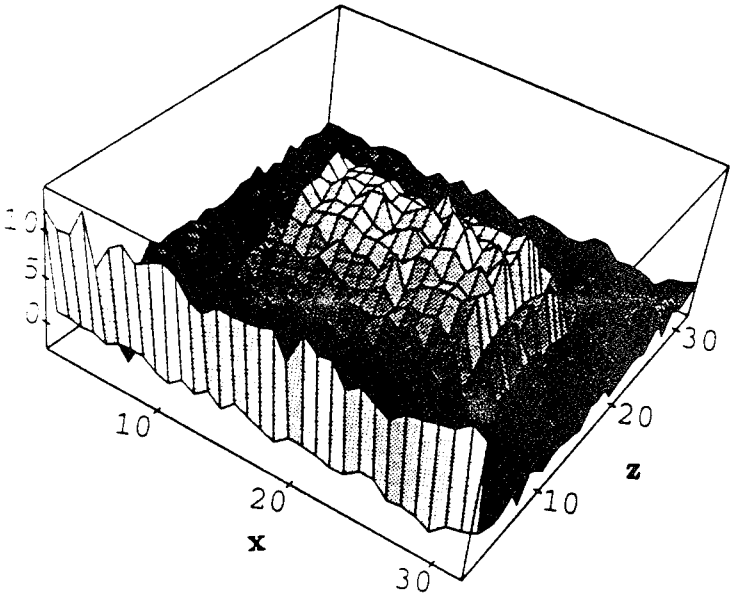


FIG. 7c

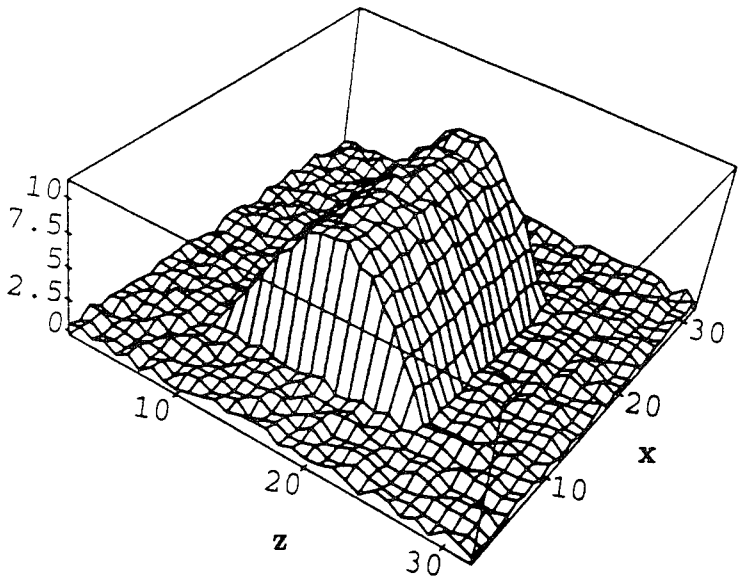


FIG. 8a

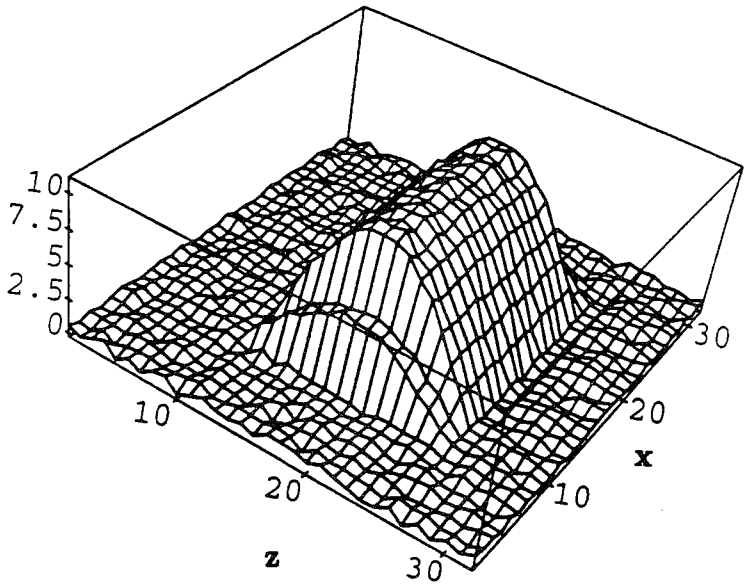


FIG. 8b

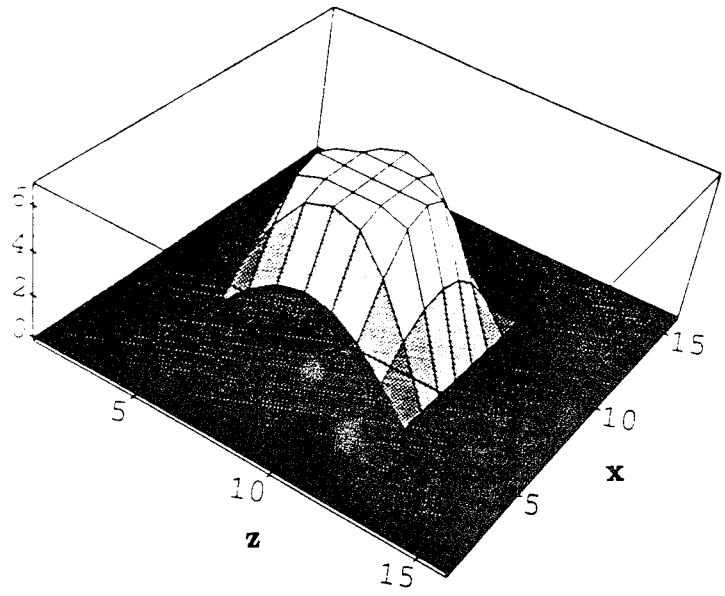


FIG. 9a

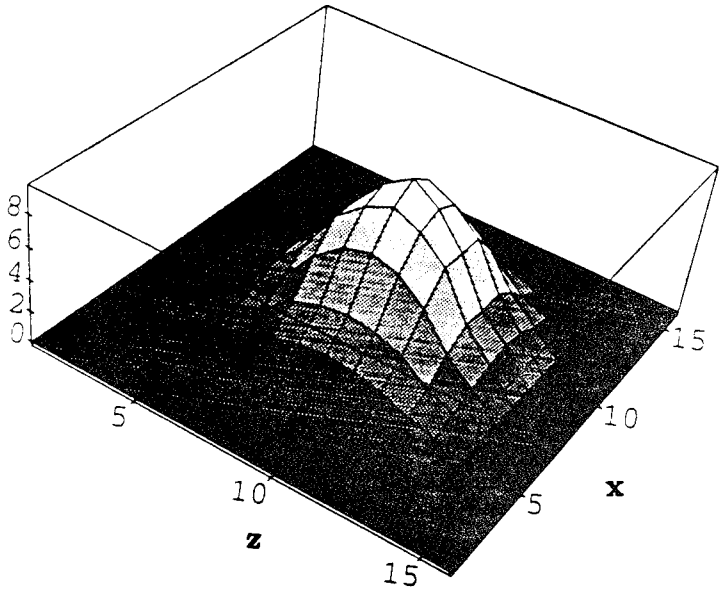


FIG. 9b

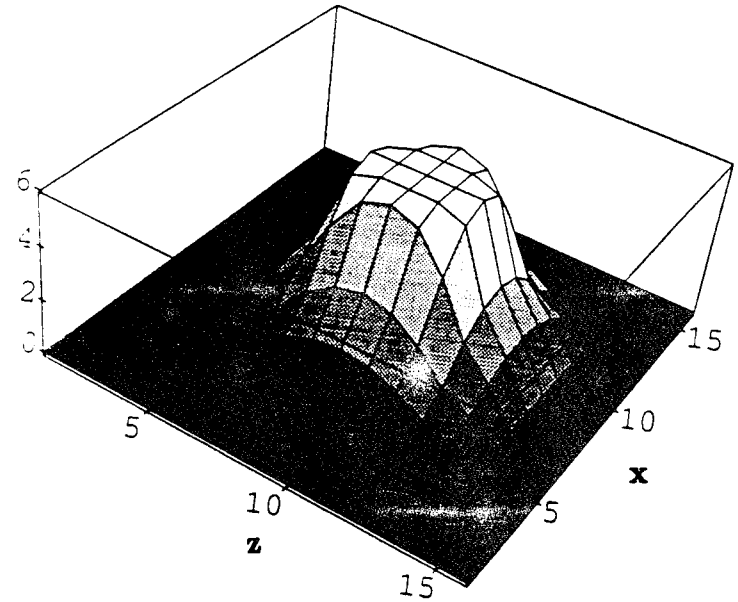


FIG. 9c

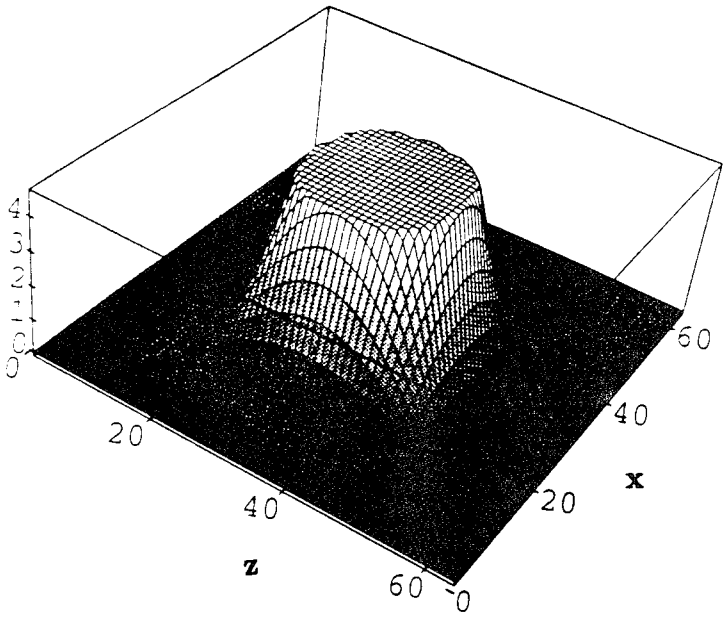


FIG. 10a

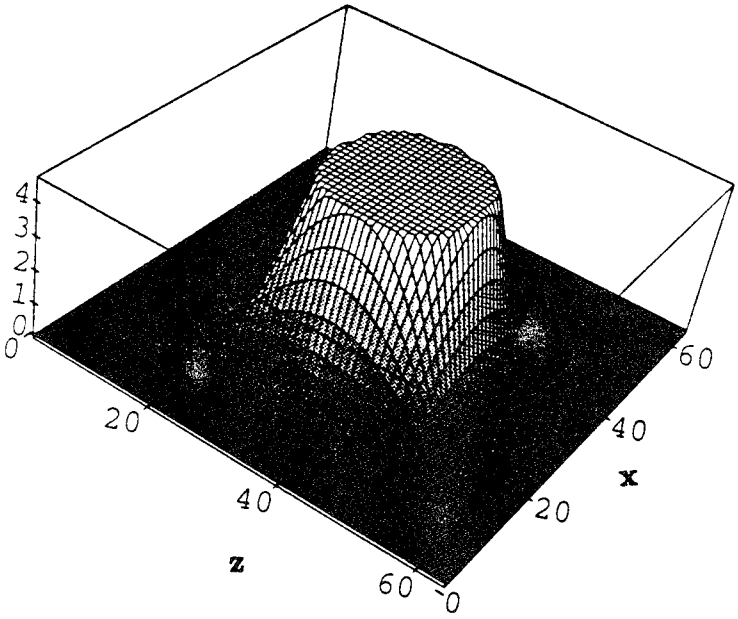


FIG. 10b

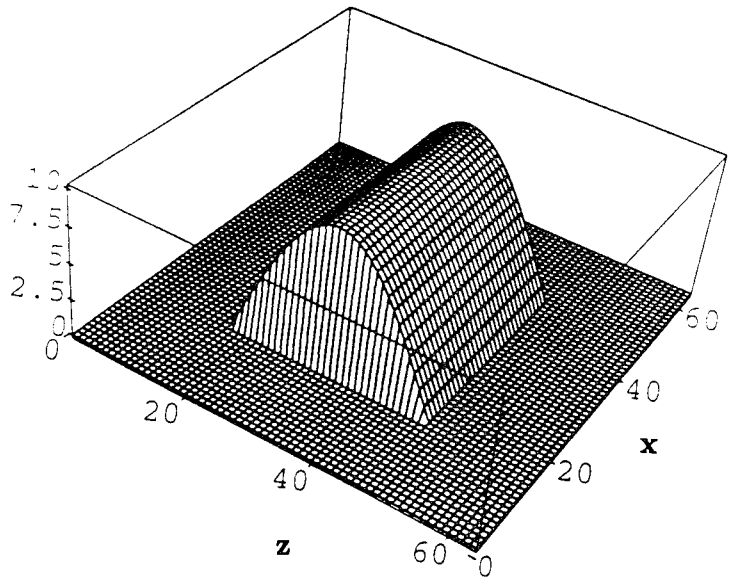


FIG. 11a

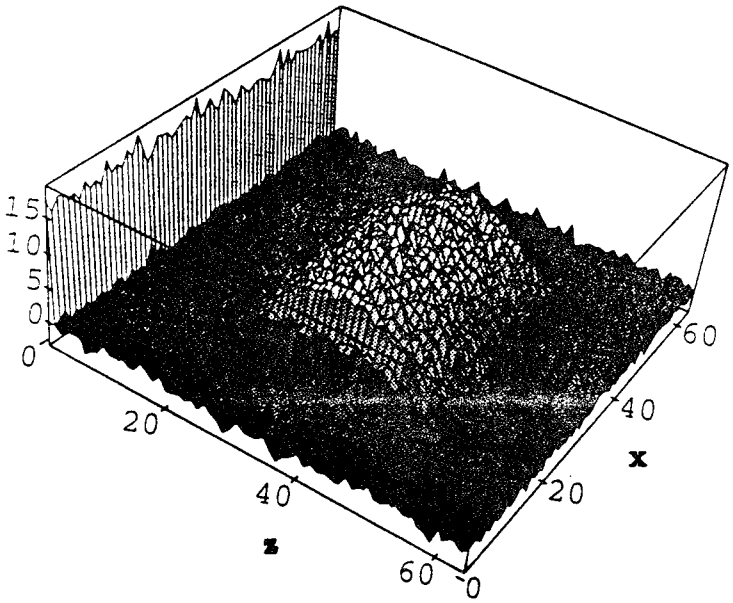


FIG. 11b

APPENDIX B

A.E. Yagle, "On the Feasibility of Impulse Reflection Response Data for the Two-Dimensional Inverse Scattering Problem," submitted to SIAM J. Appl. Math, October 1993.

This paper presents many of our most significant research results of the last year. Its contents are described in some detail in Section III.

ON THE FEASIBILITY OF IMPULSE REFLECTION RESPONSE DATA FOR THE TWO-DIMENSIONAL INVERSE SCATTERING PROBLEM

Andrew E. Yagle
Dept. of Electrical Engineering and Computer Science
The University of Michigan, Ann Arbor, MI 48109-2122
July 1993

Abstract. We reformulate the 2-D Schrodinger equation inverse scattering problem as a multichannel two-component wave system by Fourier transforming the Schrodinger equation in the lateral spatial variable. Discretization results in new 2-D layer stripping algorithms, which incorporate multichannel transmission effects; this also leads to an important new feasibility condition on impulse reflection response data. A 2-D discrete Schrodinger equation is defined, and analogous results are obtained. Numerical examples illustrate the new results.

Keywords. Multidimensional inverse scattering, numerical algorithms

AMS (MOS) subject classification. 34A55, 35P25, 35R30, 39A12

1. Introduction.

1.1 Applications and Previous Approaches. The inverse scattering problem for the Schrodinger equation in two dimensions with a time-independent, local, non-circularly symmetric potential has many applications. These applications include: (1) reconstruction of a three-dimensional (3-D) acoustic medium with density and wave speed varying in two dimensions (2-D), from surface measurement of the steady-state medium displacement response to a harmonic line source [1]; and (2) reconstruction of a 3-D electrical medium with resistivity varying in 2-D from surface measurement of the potential resulting from a line DC current source [2].

Two major approaches for exact solution of the 2-D Schrodinger equation inverse potential problem have been proposed. The first is the 2-D version of the Gel'fand-Levitan and Marchenko integral equation methods [3]. The other is the 2-D version of the layer stripping differential methods [4]. Here "exact" means that all diffraction and multiple scattering effects are included in the mathematical solution; errors in the solution will arise solely due to purely numerical effects such as discretization and roundoff. Hence all methods based on the Born (single-scattering) approximation are excluded here, since such methods, and their modifications, do not take into account *all* multiple scattering effects. No numerical implementation of the integral equation methods of [3] has been reported. The 2-D layer stripping algorithm proposed in [4] was numerically implemented successfully in [5], and shown to be robust in the presence of small noise levels.

1.2 New Results. This paper proposes two new 2-D layer stripping algorithms different from those of [4] and [5]. The new algorithms have the form of a multichannel coupled two-component wave system in which downgoing and upgoing wave matrices are scattered into each other, and where each channel corresponds to a different lateral wavenumber. This form is somewhat reminiscent of the wave systems of [6] et seq., in which downgoing and upgoing waves are again defined for 2-D scattering media. However, the wave system of this paper differs from that of [6] in three important ways: (1) it is derived for the Schrodinger equation inverse potential problem, instead of the wave equation or Maxwell's equations, as in [6]; (2) it is the basis for a computationally efficient layer stripping type algorithm, instead of a much more computationally intensive invariant imbedding algorithm,

as in [6]; and (3) it is much simpler in form than the wave system proposed in [6].

The last point is particularly important, as it leads to a central result of this paper, which is a feasibility condition on the impulse reflection data. Specifically, we show that the lateral Fourier transform of the free-surface (perfectly reflecting) impulse reflection response input must be a 2-D positive definite function in time and lateral wavenumber. This generalizes a famous result of Kunetz [7] to the 2-D case. While the Kunetz result was generalized to the multichannel case in [8], this is the first result for the 2-D inverse potential problem for the Schrodinger equation. Although our result is derived using a discrete argument, it holds for arbitrarily small discretizations.

The significance of this result is that additive noise in the data may destroy this positive definite property, and this will render the layer stripping algorithms numerically unstable. Moreover, the reputation for numerical instability of layer stripping algorithms in noise is due to infeasible data: if the data are not positive definite, then there is no scattering potential that could have produced this reflection response. We show that filtering noisy data to produce feasible data results in a numerically stable reconstruction.

1.3 Organization. This paper is organized as follows. In Section 2 we quickly review results for the 1-D Schrodinger equation inverse scattering problem. This includes the Miura transform between the Schrodinger equation and two-component wave systems, discussion of discretization vs. discrete media, and the Kunetz result [7] and its significance. In Section 3 we generalize the results of Section 2 to the 2-D case by first taking a Fourier transform in the lateral spatial variable. This results in two new types of layer stripping algorithms, which in turn solve discrete counterparts to integral equations. This is the first explicit demonstration of the direct link between differential and integral equation methods for 2-D inverse scattering; it generalizes the results of [9],[10] to the multidimensional case. Section 3 also generalizes the Kunetz result [7] to the 2-D inverse scattering problem. Section 4 produces similar results for the 2-D inverse scattering problem in which the excitation is a point source, rather than an infinite plane wave. Results for this problem are simpler than those of Section 3. Section 5 presents some numerical examples which illustrate: (1) the numerical performance of the new algorithm; (2) the significance of multiple scattering, by comparison with results using the Born approximation; (3) how

additive noise can result in infeasible data; and (4) how the infeasible data can be projected onto the subspace of feasible data, which results in a numerically stable reconstruction. Section 6 concludes with a summary and some suggestions for future research.

2. Review of 1-D Results. We quickly review some pertinent results for the continuous and discrete 1-D inverse scattering problems. These include continuous and discrete Schrodinger equations, Miura transforms, two-component wave systems, and the Krein integral equation. We also review the layer stripping/Schur algorithm and Kunetz result for the discrete problem. Section 3 will generalize all of these results to 2-D.

2.1. Continuous Problems. The 1-D Schrodinger equation inverse scattering problem is defined as follows. A wave field $u(x,k)$, where x denotes depth and k denotes frequency (for unit wave speed) or wavenumber, satisfies the Schrodinger equation

$$\left(\frac{d^2}{dx^2} + k^2 - V(x)\right) u(x,k) = 0, \tag{2.1}$$

along with the boundary conditions

$$u(x,k) = \begin{cases} e^{-ikx} + R(k)e^{ikx}, & \text{if } x \le 0; \\ T(k)e^{-ikx}, & \text{if } x \to \infty, \end{cases} \tag{2.2}$$

where $R(k)$ is known. The goal is to reconstruct the scattering potential $\{V(x), x > 0\}$ from the Fourier transform $R(k)$ of the impulse reflection response. This formulation was first applied to the inverse scattering problem of reconstructing a continuous layered acoustic medium from its impulse reflection response in [11]; there have been many other applications since then.

Now define the two-component wave system

$$\frac{d}{dx}\begin{bmatrix} D(x,k) \\ U(x,k) \end{bmatrix} = \begin{bmatrix} -ik & -r(x) \\ -r(x) & ik \end{bmatrix} \begin{bmatrix} D(x,k) \\ U(x,k) \end{bmatrix}, \tag{2.3}$$

where $D(x,k)$ and $U(x,k)$ are Fourier transforms of downgoing and upgoing waves, respectively, and $r(x)$ is the reflectivity function. The boundary conditions for (2.3) are

$$D(x,k) = e^{-ikx}; \quad U(x,k) = R(k)e^{ikx}, \quad x \le 0 \tag{2.4a}$$

$$D(x,k) = T(k)e^{-ikx}; \quad U(x,k) = 0, \quad x \to \infty. \tag{2.4b}$$

The inverse scattering problem for the wave system (2.3) is to reconstruct the reflectivity function $\{V(x), x > 0\}$ from $R(k)$. Again, this formulation has been applied to the inverse scattering problem of reconstructing a continuous layered acoustic medium from its impulse reflection response (see the reference list of [9]).

The Schrodinger equation (2.1) can be transformed into the two-component wave system (2.3) using the Miura transform [9]. Given the potential $V(x)$, define the reflectivity function $r(x)$ as the solution to

$$V(x) = r^2(x) - dr(x)/dx; \quad u(x,k) = D(x,k) + U(x,k), \tag{2.5}$$

where $r(0)$ is assumed to be known (since $x = 0$ denotes the surface of the scattering medium). Then the inverse scattering problem defined by (2.1)-(2.2) is equivalent to the inverse scattering problem defined by (2.3)-(2.4).

2.2. Derivation of Layer Stripping Algorithm. We review the derivation of the layer stripping algorithm for the two-component wave system inverse scattering problem (2.3)-(2.4). The point here is to compare the discretized continuous layer stripping algorithm with the Schur algorithm in Section 2.3.

In the time domain (2.3) becomes the pair of equations

$$\left(\frac{\partial}{\partial x} + \frac{\partial}{\partial t}\right) \check{D}(x,t) = -r(x)\check{U}(x,t) \tag{2.6a}$$

$$\left(\frac{\partial}{\partial x} - \frac{\partial}{\partial t}\right) \check{U}(x,t) = -r(x)\check{D}(x,t) \tag{2.6b}$$

where $\check{D}(x,t) = \int_{-\infty}^{\infty} D(x,k)e^{ikt}dk$ is the inverse Fourier transform of $D(x,k)$, and similarly for $\check{U}(x,t)$. $\check{D}(x,t)$ and $\check{U}(x,t)$ are clearly waves, since (2.6) describe quantities that propagate in increasing and decreasing depth x as t increases. The reflectivity function $r(x)$ describes how much of each wave is reflected into the other wave at each depth x.

Since the impulse reflection response $\check{R}(t)$ is causal, it is clear that $\check{D}(x,t)$ and $\check{U}(x,t)$ have the forms

$$\check{D}(x,t) = \delta(t-x) + \tilde{D}(x,t)1(t-x) \tag{2.7a}$$

$$\check{U}(x,t) = \tilde{U}(x,t)1(t-x) \tag{2.7b}$$

where $\tilde{D}(x,t)$ and $\tilde{U}(x,t)$ are the smooth parts of $\check{D}(x,t)$ and $\check{U}(x,t)$ (both of which jump at $t = x$), and where $1(\cdot)$ is the unit step or Heaviside function. Inserting (2.7) into (2.6) and using a propagation of singularities argument (this amounts to equating coefficients of $\delta(t-x)$) yields

$$\left(\frac{\partial}{\partial x} + \frac{\partial}{\partial t}\right)\tilde{D}(x,t) = -r(x)\tilde{U}(x,t) \tag{2.8a}$$

$$\left(\frac{\partial}{\partial x} - \frac{\partial}{\partial t}\right)\tilde{U}(x,t) = -r(x)\tilde{D}(x,t) \tag{2.8b}$$

$$r(x) = 2\tilde{U}(x,x^+). \tag{2.8c}$$

We now discretize depth $x = m\Delta$ and time $t = n\Delta$ to integer multiples of some discretization length Δ. Since the wave speed is unity, depth and time have the same Δ. Using forward differences, (2.8) discretizes to

$$\begin{bmatrix}\tilde{D}(x+\Delta,t)\\ \tilde{U}(x+\Delta,t)\end{bmatrix} = \begin{bmatrix}1 & -r(x)\Delta\\ -r(x)\Delta & 1\end{bmatrix}\begin{bmatrix}\tilde{D}(x,t-\Delta)\\ \tilde{U}(x,t+\Delta)\end{bmatrix} \tag{2.9a}$$

$$r(x)\Delta = \tilde{U}(x,x)/\tilde{D}(x,x-2\Delta) \tag{2.9b}$$

$$\tilde{D}(0,t) = 1; \quad \tilde{U}(0,t) = \int_{-\infty}^{\infty} R(k)e^{ikt}dk. \tag{2.9c}$$

Equations (2.9) are the layer stripping algorithm for solving the inverse scattering problem specified by (2.3)-(2.4). Note that the factor of 2 in (2.8c) disappears in (2.9b), which follows from (2.9a) by setting $t = x$ and noting that $\tilde{U}(x+\Delta, x-\Delta) = 0$ by causality. Once $r(x)$ has been reconstructed, the Schrodinger equation inverse scattering problem (2.1)-(2.2) is solved by computing $V(x)$ from $r(x)$ using (2.5). Note that the differential equation (2.5) need never be solved–(2.5) is a formula for computing $V(x)$ from $r(x)$.

2.3. Discrete Problems. The discrete 1-D Schrodinger equation inverse scattering problem is defined as follows. A wave field $u(i,j)$, where i and j are depth and time indices, satisfies the discrete Schrodinger equation

$$u(i+1,j) + u(i-1,j) - u(i,j+1) - u(i,j-1) = V(i)u(i-1,j), \tag{2.10}$$

along with the boundary conditions

$$u(i,j) = \begin{cases}\delta(j-i) + R(j+i), & \text{if } i \le 0;\\ T(j-i), & \text{if } i \to \infty.\end{cases} \tag{2.11}$$

The discrete inverse scattering problem is to reconstruct $V(i)$ from $R(j)$. This problem has been formulated and discussed in [12] and [13]. Note that the left side of (2.10) can be viewed as a discrete form of the inverse Fourier transform of the left side of (2.1). If the indices are scaled to be multiples of Δ, then (2.10) approaches the inverse Fourier transform of (2.1) as $\Delta \to 0$.

The discrete two-component wave system is defined as

$$\begin{bmatrix} D(i+1,j) \\ U(i+1,j) \end{bmatrix} = \frac{1}{\sqrt{1-r(i)^2}} \begin{bmatrix} 1 & -r(i) \\ -r(i) & 1 \end{bmatrix} \begin{bmatrix} D(i,j-1) \\ U(i,j+1) \end{bmatrix}, \tag{2.12}$$

where $D(i,j)$ and $U(i,j)$ are the downgoing and upgoing waves just below the i^{th} interface at time j. Note the time shifts required by the unit time required to propagate through each layer of unit thickness at unit wave speed. The boundary conditions are

$$D(i,j) = \delta(j-i); \quad U(i,j) = R(j-i), \quad i \leq 0 \tag{2.13a}$$

$$D(i,j) = T(j-i); \quad U(i,j) = 0, \quad i \to \infty. \tag{2.13b}$$

The discrete inverse scattering problem is to reconstruct $r(i)$ from $R(j)$. This problem has been analyzed in detail in [10] and earlier references listed in [10]; $r(i)$ is the interface reflection coefficient.

The discrete Schrodinger equation (2.10) can be transformed into the two-component wave system (2.12) using a discrete form of the Miura transform. Given the potential $V(i)$, define the reflection coefficient $r(i)$ as the solution to (compare to (2.5))

$$V(i) = r(i)r(i-1) - (r(i) - r(i-1)); \quad u(i-1,j) = D(i,j-1) + U(i,j+1), \tag{2.14}$$

where $r(0)$ is assumed to be known (since $i = 0$ denotes the surface of the scattering medium). Then the inverse scattering problem defined by (2.10)-(2.11) is equivalent to the inverse scattering problem defined by (2.12)-(2.13). The transformation (2.14) was first employed in [14] to derive the so-called split Levinson and Schur algorithms. Its application to inverse scattering as a discrete form of the Miura transform was first noted in [15]. Note again that (2.14) is a formula for computing $V(i)$ from $r(i)$, not a difference equation.

2.4. Schur Algorithm. The discrete inverse scattering problem defined by (2.12)-(2.13) can be solved very easily using the Schur algorithm [9],[10], a famous signal processing algorithm in the solution of Toeplitz systems of equations. The Schur algorithm is identical to the layer stripping algorithm (2.9), with $\Delta = 1$ and independent variables (x,t) replaced with (i,j). Note that $D(i,j)$ and $U(i,j)$ are zero for odd values of $i+j$. The reason for this is physically quite clear: since waves require two time units to propagate through a layer and return, waves can arrive at a given interface only for alternate values of time j.

The major point here, whose importance cannot be overemphasized, is that the success of the layer stripping algorithm is due entirely to its equivalence to the Schur algorithm. The algorithm (2.9), which is ostensibly a discretization of continuous equations for solving a continuous problem, is in fact solving the explicitly discrete problem (2.12)-(2.13) ***exactly***, without approximation. Furthermore, the Schur algorithm has excellent numerical stability properties because it reconstructs exactly the lossless medium described by (2.12).

The two-component wave system (2.12) is lossless and well-behaved numerically as long as $|r(n)| < 1$ [10]. The necessity of $|r(n)| < 1$ is clear from the transmission loss factor $\sqrt{1-r(i)^2}$ in (2.12); sufficiency follows from the fact that (2.12) can be rearranged into the scattering form

$$\begin{bmatrix} D(i+1,j) \\ U(i,j+1) \end{bmatrix} = \begin{bmatrix} \sqrt{1-r(i)^2} & -r(i) \\ r(i) & \sqrt{1-r(i)^2} \end{bmatrix} \begin{bmatrix} D(i,j-1) \\ U(i+1,j) \end{bmatrix}, \tag{2.15}$$

in which the single-layer scattering matrix is orthogonal. Note that for discrete lossy transmission lines and discrete layered acoustic media, the interface reflection coefficient is $r(i) = (Z(i+1) - Z(i))/(Z(i+1) + Z(i))$, where $Z(i)$ is the impedance of the i^{th} layer, so that $|r(n)| < 1$ is guaranteed.

2.5. Kunetz Condition. Now suppose that the top interface is a free surface (perfect reflector), and that the probing impulse in (2.13a) is introduced just below this surface (so that now $D(i,j) = \delta(j-i) + R(j-i)$). In this case, the Schur algorithm

computes the reflection coefficients associated with the Toeplitz system of equations

$$\begin{bmatrix} 1 & R(1) & \cdots & R(n) \\ R(1) & 1 & \cdots & R(n-1) \\ \vdots & \vdots & \ddots & \vdots \\ R(n) & R(n-1) & \cdots & 1 \end{bmatrix} \begin{bmatrix} F_n^0 - G_n^n \\ \vdots \\ F_n^n - G_n^0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ t_n \end{bmatrix}, \tag{2.16}$$

where the $R(j)$ are the impulse reflection response defined in (2.13a), the F_n^i and G_n^i are elements of the matrix Green's function of (2.12) (see [10]) and $t_n = \prod \sqrt{1 - r(i)^2}$ is the transmission loss through n layers. The reflection coefficients can be recovered by $r(n) = G_n^0 - F_n^n$.

Eq. (2.16) can be interpreted in two different ways: (1) as a discrete counterpart of the Krein integral equation (see [9],[10], or scale i and j by Δ and let $\Delta \to 0$); or (2) as a set of Yule-Walker equations for computing the least-squares autoregressive prediction filter coefficients of a discrete-time zero-mean stationary random process with unit variance and covariance lags $R(i)$. The transmission loss t_n through n layers in (2.16) is analogous to the rms prediction error of an n^{th}-order filter.

Since the covariance lags of a stationary random process must form a positive definite sequence, the following result, attributed to Kunetz [7], is not surprising. Let $R(n)$ be the impulse response of a discrete lossless layered medium with a free surface. Then the two-sided sequence $\{\ldots R(2), R(1), 1, R(1), R(2) \ldots\}$ is positive definite. In fact, it is the autocorrelation of the transmission response $T(j)$ defined in (2.13b). This result is derived algebraically in [7]. Furthermore, a well-known result in linear prediction theory [16] states that the reflection coefficients have the property $|r(n)| < 1$ if and only if the system matrix (2.16) is positive definite, i.e., the $R(j)$ and their time reversal form a positive definite sequence.

The significance of this result in applying the layer stripping/Schur algorithm (2.9) is as follows. If the impulse reflection response data are corrupted by noise so that they no longer constitute a positive definite sequence, then the algorithm will fail. This is appropriate, since such data are *infeasible* in that there is no lossless medium that could give rise to such data. This is why layer stripping algorithms have the reputation of being unstable in noise: they are being fed infeasible data! If the noisy data are processed so

that it forms a positive definite (although still noisy) sequence, the algorithm will behave well numerically. Of course, it will reconstruct the medium associated with the noisy data, not the actual medium, but it will not diverge.

In the next section we generalize all of these results to the 2-D case.

3. New 2-D Results for Plane-Wave Excitation.

3.1. Continuous Problem. The 2-D Schrodinger equation inverse scattering problem is defined as follows. The problem is defined in (x, z) space, where x is lateral position and z is depth, increasing downward from the surface $z = 0$. The wave field $\hat{u}(x, z, k)$ satisfies the 2-D Schrodinger equation

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial z^2} + k^2 - \hat{V}(x, z)\right) \hat{u}(x, z, k) = 0, \tag{3.1}$$

where the potential $\hat{V}(x, z)$ is real-valued and smooth. It is also assumed that $\hat{V}(x, z)$ does not induce bound states; a sufficient condition for this is for $\hat{V}(x, z)$ to be non-negative.

In separate experiments, the medium is probed at various angles of incidence θ to the vertical by an impulsive plane wave $\delta(t - x \sin\theta - z \cos\theta)$ which at the point $x = 0$ passes through the surface $z = 0$ at time $t = 0$ and induces scattering due to $\hat{V}(x, z)$ for $t > 0$. A Fourier transform taking time t into frequency ω results in $e^{-i\omega(x \sin\theta - z\cos\theta)}$ as the excitation. Since each experiment is performed separately, we may define the wavenumbers $k = \omega \cos\theta$ and $k_x = \omega \sin\theta$; the excitation for each experiment then becomes $e^{-ikz}e^{-ik_x x}$. Note that for normal incidence $\theta = 0$ this reduces to e^{-ikz} (compare to (2.2)). Note also that the effective medium wave speed c_0 defined by $k = \omega/c_0$ varies for each experiment, but is constant for a given experiment.

The data are the reflection responses $R(k, x; k_x)e^{ikz}$ in the direction of decreasing z resulting from the plane waves $e^{-ikz}e^{-ik_x x}$ (compare to (2.2)). Note this is the backscattered field for all x in the infinite half space $z \leq 0$. The goal is to reconstruct the potential $\hat{V}(x, z)$ from the reflection responses $R(k, x; k_x)$. In fact, Section 4 will show that the reflection response to a single impulsive point source is sufficient, but this more general formulation is also of interest.

3.2. Applications. We now quickly review two applications of this problem. First,

consider the problem of reconstructing a 3-D inhomogeneous acoustic medium whose density $\rho(x,z)$ and wave speed $c(x,z)$ are smooth functions of depth z and lateral position x. The medium is bounded by a free (pressure-release) surface $z = 0$. The density ρ_0 and wave speed c_0 for $z < 0$ and $z \to \infty$ are known. The medium is probed with cylindrical harmonic waves, at two frequencies ω_1 and ω_2, from a harmonic line source extending along the x-axis, and the sinusoidal steady-state vertical acceleration $\hat{a}(x,y,z=0;\omega_i)$ of the medium at the free surface $z = 0$ is measured. The goal is to reconstruct $\rho(x,z)$ and $c(x,z)$ from the measurements $\hat{a}(x,y,z=0;\omega_i), i = 1,2$.

This problem can be formulated as a 2-D Schrodinger equation inverse potential problem by Fourier transforming the basic acoustic equations with respect to time and the other lateral variable y. Details are given in both [1] and [4]. Here we merely note that in the Schrodinger equation (2.1) the wave field $\hat{u}(x,z,k)$ is pressure divided by $\rho(x,z)^{1/2}$, the wavenumber $k^2 = \omega_i^2/c_0^2 - k_y^2$, and the potential $\hat{V}(x,z;\omega_i)$ is

$$\hat{V}(x,z;\omega_i) = \left(\frac{\omega_i^2}{c_0^2}\right)\left(1 - \frac{c_0^2}{c(x,z)^2}\right) + \rho(x,z)^{1/2}\nabla^2(\rho(x,z)^{-1/2}). \tag{3.2}$$

It is clear that performing this experiment for two different frequencies $\omega_i, i = 1,2$ will allow $\rho(x,z)$ and $c(x,z)$ to be computed from (3.2). The wave field is zero at the free surface $z = 0$; its gradient is the medium acceleration $\rho(x,0)^{1/2}\hat{a}(x,y,z=0;\omega_i), i = 1,2$.

The second application is the inverse resistivity problem of reconstructing a 3-D inhomogeneous electrical medium whose resistivity $\rho(x,z)$ is a smooth function of x and z over a bounded region. The medium is probed with current from a line DC current source extending along the x-axis, and the electrical potential $v(x,y,z=0)$ induced on the surface $z = 0$, assumed to be a perfect insulator, is measured. The goal is to reconstruct the resistivity $\rho(x,z)$ from the measurements of electrical potential $v(x,y,z=0)$. Note that for both applications, the response to a *line* source may be found by superposition of the responses due to *point* sources along the x-axis.

This problem can be formulated as a 2-D Schrodinger equation inverse potential problem by Fourier transforming Ohm's and Kirchoff's current laws with respect to the other lateral variable y. Details are given in [2]. Here we merely note that in the Schrodinger equation (3.1) the wave field $\hat{u}(x,z,k)$ is now the inverse Laplace transform

of the Fourier transform of electrical potential divided by $\rho(x,z)^{1/2}$, and the scattering potential $\hat{V}(x,z) = \rho(x,z)^{1/2}\nabla^2(\rho(x,z)^{-1/2})$.

3.3. Reformulation of (3.1) as a Two-Component Wave System. Taking the Fourier transform of (3.1) in the lateral spatial variable x yields

$$\left(\frac{d^2}{dz^2} + k^2 - k_x^2\right) u(z,k,k_x) = \int_{-\infty}^{\infty} V(z,k_x - k_x')u(z,k,k_x')dk_x', \tag{3.3}$$

where

$$u(z,k,k_x) = \int_{-\infty}^{\infty} \hat{u}(x,z,k)e^{-ik_x x}dx; \quad V(z,k_x) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{V}(x,z)e^{-ik_x x}dx. \tag{3.4}$$

The multiplication by k_x^2 in (3.3) will induce numerical instability. Hence we replace k_x^2 with $F(k_x^2)$, where $F(k_x^2)$ is a function that windows k_x^2 to zero for large $|k_x|$. One possible choice of $F(k_x^2)$ is

$$F(k_x^2) = \begin{cases} k_x^2, & \text{if } |k_x| < K; \\ 0, & \text{otherwise} \end{cases} \tag{3.5}$$

for some cutoff wavenumber K. This is reminiscent of the clipped filter used in the filtered back-projection procedure for inverting the Radon transform.

Replacing k_x^2 with $F(k_x^2)$, (3.3) may be rewritten as

$$\left(\frac{d^2}{dz^2} + k^2\right) u(z,k,k_x) = \int_{-\infty}^{\infty} \left(F(k_x^2)\delta(k_x - k_x') + V(z,k_x - k_x')\right) u(z,k,k_x')dk_x'. \tag{3.6}$$

To clarify the 2-D form of the Miura transform (2.5), we now discretize k_x into integer multiples of an arbitrarily small Δ. This results in

$$\left(\frac{d^2}{dz^2} + k^2 - \mathbf{V}\right) \mathbf{u}(z,k) = \mathbf{0}, \tag{3.7}$$

where $\mathbf{u}(z,k)$ is a matrix whose $(m,n)^{th}$ element is $u(z,k,m\Delta)$ associated with the experiment with excitation $e^{-ikz}e^{-inx\Delta}$ and $\mathbf{V}$ is a Toeplitz-plus-diagonal matrix with $(i,j)^{th}$ element

$$\mathbf{V}_{(i,j)} = F((i\Delta)^2)\delta(i-j) + V(z,(i-j)\Delta)\Delta. \tag{3.8}$$

Note that the indices in (3.8) run from $-N$ to N, where N is an arbitrarily large integer, and that $\mathbf{V}$ is Hermitian, since $\hat{V}(x,z)$ is real.

Now define the matrix reflectivity function $\mathbf{r}$ by (compare to (2.5))

$$\mathbf{V} = \mathbf{r}^2 - d\mathbf{r}/dz; \quad \mathbf{u}(z,k) = (\mathbf{D}+\mathbf{U})(z,k), \tag{3.9}$$

where $\mathbf{r}(0)$ is assumed to be known (since $z = 0$ denotes the surface of the scattering medium). Then (3.7) can be transformed into the multichannel two-component wave system (compare to (2.3))

$$\frac{d}{dz}\begin{bmatrix}\mathbf{D}(z,k)\\ \mathbf{U}(z,k)\end{bmatrix} = \begin{bmatrix}-ikI & -\mathbf{r}(z)\\ -\mathbf{r}(z) & ikI\end{bmatrix}\begin{bmatrix}\mathbf{D}(z,k)\\ \mathbf{U}(z,k)\end{bmatrix} \tag{3.10}$$

in which $\mathbf{D}(z,k)$, $\mathbf{U}(z,k)$, $\mathbf{r}(z)$, and $\mathbf{R}(k)$ (see (3.11)) are all $N \times N$ matrices. For each of these matrices, different columns correspond to different experiments, while different rows correspond to different channels for a given experiment.

The boundary conditions for (3.10) are (compare to (2.4))

$$\mathbf{D}(z,k) = e^{-ikz}I; \quad \mathbf{U}(z,k) = \mathbf{R}(k)e^{ikz}, \quad z \le 0 \tag{3.11a}$$

$$\mathbf{D}(z,k) = \mathbf{T}(k)e^{-ikz}; \quad \mathbf{U}(z,k) = \mathbf{0}, \quad z \to \infty \tag{3.11b}$$

$$\mathbf{R}(k)_{(m,n)} = \int_{-\infty}^{\infty} R(k,x;n\Delta)e^{-imx\Delta}\Delta\, dx \tag{3.11c}$$

where $R(k,x;k_x)e^{ikz}$ is the reflection response to $e^{-ikz}e^{-ik_x x}$. The Fourier transform of $e^{-ikz}e^{-ik'_x x}$ is $e^{-ikz}\delta(k_x - k'_x)$, so each excitation $e^{-ikz}e^{-ik'_x x}$ for (3.1) excites a single channel in (3.10). So the $(m,n)^{th}$ element of $\mathbf{D}(z,k)$ or $\mathbf{U}(z,k)$ is the downgoing or upgoing wave at depth z in the m^{th} channel, resulting from a downgoing impulse e^{-ikz} in the n^{th} channel only.

Hence the 2-D Schrodinger equation inverse scattering problem (3.1) can be transformed into the multichannel two-component wave system inverse scattering problem (3.10)-(3.11), by way of the multichannel Schrodinger equation (3.7). Once $\mathbf{r}(z)$ has been reconstructed, the Schrodinger equation inverse scattering problem (3.1) is solved by computing $\mathbf{V}(z)$ from $\mathbf{r}(z)$ using (3.9). Note the differential equation (3.9) need not be solved.

3.4. Discretization. The multichannel two-component wave system (3.10) has the same form as the scalar two-component wave system (2.3). The development in Section 2.2 generalizes directly to the multichannel case; the layer stripping algorithm for the

multichannel two-component wave system (3.10), and hence for the Schrodinger equation (3.1), is (compare to (2.9))

$$\begin{bmatrix} \tilde{\mathbf{D}}(z+\Delta,t) \\ \tilde{\mathbf{U}}(z+\Delta,t) \end{bmatrix} = \begin{bmatrix} 1 & -\mathbf{r}(z)\Delta \\ -\mathbf{r}(z)\Delta & 1 \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{D}}(z,t-\Delta) \\ \tilde{\mathbf{U}}(z,t+\Delta) \end{bmatrix} \tag{3.12a}$$

$$\mathbf{r}(z)\Delta = \tilde{\mathbf{U}}(x,x)\tilde{\mathbf{D}}^{-1}(x,x-2\Delta) \tag{3.12b}$$

$$\tilde{\mathbf{D}}(0,t) = I; \quad \tilde{\mathbf{U}}(0,t) = \int_{-\infty}^{\infty} \mathbf{R}(k)e^{ikt}dk, \tag{3.12c}$$

where Δ is discretization length for depth z and time t, as in (2.9).

It is worth examining the above discretizations from a digital signal processing perspective. We now recall some points made in [5] about actual numerical implementation of discretized equations such as (3.12). Let Δ_x be the discretization length for z and t in (3.12), and let Δ_k be the discretization length for k_x in (3.7). Note that Δ and Δ_k have reciprocal units. First, note that Δ_k should be much less than half the reciprocal of the total lateral extent of interest, to avoid aliasing. For example, if the potential has finite support $-L_x/2 < x < L_x/2$ in x, L_x would be the lateral extent of interest.

Second, note that the discretized functions $\tilde{\mathbf{D}}(z+\Delta,t)$, etc. cannot be regarded as sampled versions of the continuous functions $\tilde{\mathbf{D}}(z,t)$, etc., even if sampling is performed above the Nyquist rate, since the convolution in (3.6) (which becomes the products of Toeplitz matrices with columns of other matrices in (3.7)) mixes the k_x. Even if the inverse potential problem is regularized by assuming that $V(z,k_x)$ is bandlimited in z and zero for $|k_x| > K$ for some K, it is clear that $u(z,k,k_x)$, etc. will NOT have similar properties. Imposing a bandlimited in k_x condition at each recursion will lead to errors, since the missing high k_x will cause errors for low k_x due to the mixing of the k_x.

This leads to the question of what the discretized functions mean, and how the convolutions in k_x should be performed. It should be noted that similar questions arise in integral equation methods. One possible interpretation is to perform a periodic extension in k of all quantities. The period in k should be $1/\Delta$; K in (3.5) should then be half this. It is clear by induction that if all quantities at depth $z = n\Delta$ are periodic in k, then all quantities at depth $z = (n+1)\Delta$ will also be periodic in k.

This creates two advantages: (1) the infinite linear convolutions becomes finite cyclic convolutions; and (2) the discrete Fourier transform may be used to perform all Fourier

transforms. In terms of (3.7), $\mathbf{V}(z)$ is now a circulant matrix. Since periodicity in one Fourier domain is equivalent to discreteness in the other Fourier domain, the problem has effectively been discretized laterally as well as vertically: the quantites propagated in (2.11) are not samples of a bandlimited function, but actual discrete values. As $\Delta \to 0$, the situation approaches the continuous problem.

3.5. Discrete Problem. Another approach is to formulate and solve an explicitly discrete 2-D Schrodinger equation inverse scattering problem. Here we present for the first time such a problem, generalizing the results reviewed in Section 2.3.

We define the discrete 2-D Schrodinger equation inverse scattering problem as follows. A wave field $u(i,j,n)$, where i is depth, j is lateral position, and n is time, satisfies the 2-D discrete Schrodinger equation

$$u(i+1,j,n)+u(i-1,j,n)+u(i-1,j+1,n)+u(i-1,j-1,n)-2u(i-1,j,n)$$

$$-u(i,j,n+1)-u(i,j,n-1)=V(i-1,j)u(i-1,j,n). \tag{3.13}$$

It is clear that if the indices in (3.13) are scaled to be multiples of Δ then (3.13) becomes the time-domain form of the continuous Schrodinger equation (3.1) as $\Delta \to 0$. Note that the differences corresponding to $\partial^2/\partial x^2$ are shifted in depth z from the differences corresponding to $\partial^2/\partial z^2$ and $\partial^2/\partial t^2$. This is necessary to obtain the multichannel two-component wave system below.

A discrete-time Fourier transform of (3.13) taking j into k_x gives (compare to (3.3))

$$\hat{u}(i+1,n,k_x)+\hat{u}(i-1,n,k_x)-\hat{u}(i,n+1,k_x)-\hat{u}(i,n-1,k_x)$$

$$=(2-e^{ik_x}-e^{-ik_x})\hat{u}(i-1,n,k_x)+\sum_{m=-\infty}^{\infty}\hat{V}(i-1,k_x-m)\hat{u}(i-1,n,m) \tag{3.14a}$$

$$\hat{u}(i,n,k_x)=\sum_{j=-\infty}^{\infty}u(i,j,n)e^{-ik_xj};\quad \hat{V}(i,k_x)=\frac{1}{2\pi}\sum_{j=-\infty}^{\infty}V(i,j)e^{-ik_xj}. \tag{3.14b}$$

The definitions (3.14b) should be compared with (3.4). Note that

$$2-e^{ik_x}-e^{-ik_x}=2(1-\cos k_x)\approx k_x^2,\quad k_x\to 0 \tag{3.15}$$

which also shows how the discrete 2-D Schrodinger equation (3.13) reduces to the continuous 2-D Schrodinger equation (3.1). Also note that $2(1-\cos k_x)$ can be regarded as a choice of $F(k_x^2)$ different from (3.5).

The boundary conditions for (3.14) are (compare to (3.11))

$$\hat{u}(i,n,k_x;k_x') = \begin{cases} \delta(n-i)\delta(k_x-k_x') + R(n+i,k_x;k_x'), & \text{if } i \le 0; \\ T(n-i,k_x;k_x'), & \text{if } i \to \infty. \end{cases} \tag{3.16}$$

In (3.16) $R(n,k_x;k_x')$ is the reflection response to an impulsive plane wave that excites the lateral wavenumber k_x' only. This could be measured for an actual medium as discussed earlier. The discrete 2-D inverse scattering problem is to reconstruct $\hat{V}(i,k_x)$ from $R(n,k_x;k_x')$.

3.6. Reformulation of (3.14) as a Two-Component Wave System. As in Section 3.3, we discretize k_x into integer multiples of Δ. Eq. (3.14) can then be rewritten as (compare to (3.7))

$$\mathbf{u}(i+1,n) + \mathbf{u}(i-1,n) - \mathbf{u}(i,n+1) - \mathbf{u}(i,n-1) = \mathbf{V}(i-1)\mathbf{u}(i-1,n) \tag{3.17}$$

where $\mathbf{u}(i,n)$ is a matrix whose $(j,k)^{th}$ element is $u(i,n,j\Delta;k\Delta)$ and $\mathbf{V}(i)$ is a Toeplitz-plus-diagonal matrix with $(j,k)^{th}$ element (using (3.15); compare to (3.8))

$$\mathbf{V(i)}_{(j,k)} = 2(1-\cos(j\Delta))\delta(j-k) + \hat{V}(i,(j-k)\Delta). \tag{3.18}$$

Again the indices in (3.18) run from $-N$ to N, where N is an arbitrarily large integer, and $\mathbf{V}$ is Hermitian, since $V(i,j)$ is real.

The matrix discrete Schrodinger equation (3.18) represents the discrete 2-D Schrodinger equation (3.13), just as the matrix continuous Schrodinger equation (3.7) represents the continuous 2-D Schrodinger equation (3.1). The discrete Miura transform (2.14) generalizes to the multichannel case as follows [17]. Given the matrix potential $\mathbf{V}(i)$ defined in (3.18), define the matrix reflection coefficient $\mathbf{r}(i)$ as the solution to

$$\mathbf{V}(i) = I - \mathbf{T}(i)(I+\mathbf{r}(i))(I-\mathbf{r}(i-1))\mathbf{T}^{-1}(i) \tag{3.19a}$$

$$\mathbf{T}(i) = \prod_{j=1}^{i-1} (I-\mathbf{r}(j)^2)^{1/2} \tag{3.19b}$$

$$\mathbf{u}(i,n) = \mathbf{T}(i)(I - \mathbf{r}(i-1))^{-1}(\mathbf{D}(i,n) + \mathbf{U}(i,n)). \tag{3.19c}$$

Then the matrix discrete Schrodinger equation (3.18) is equivalent to the discrete multichannel two-component wave system (compare to (2.12),(3.10))

$$\begin{bmatrix} \mathbf{D}(i+1,j) \\ \mathbf{U}(i+1,j) \end{bmatrix} = \begin{bmatrix} (I - \mathbf{r}(i)^2)^{-1/2} & 0 \\ 0 & (I - \mathbf{r}(i)^2)^{-1/2} \end{bmatrix} \begin{bmatrix} I & -\mathbf{r}(i) \\ -\mathbf{r}(i) & I \end{bmatrix} \begin{bmatrix} \mathbf{D}(i,j-1) \\ \mathbf{U}(i,j+1) \end{bmatrix}. \tag{3.20}$$

As in the continuous system (3.10), the bold quantities are all $N \times N$ matrices. For each of these matrices, different columns correspond to different experiments, while different rows correspond to different channels for a given experiment. And the inverse scattering problem defined by (3.14)-(3.16) is equivalent to the inverse scattering problem defined by (3.20) and (3.16).

Several comments are in order here:

1. In the scalar case (3.19) reduces to (2.14) and (3.20) reduces to (2.12), as expected. However, the generalization of (2.14) to (3.19) is not at all obvious, due to the matrix transmission factor $\mathbf{T}(i)$;
2. The transmission factor $\mathbf{T}(i)$ becomes a scalar factor in (2.12) which cancels out in the layer stripping/Schur algorithm in (2.9b). This is why the 1-D layer stripping algorithm is identical to the Schur algorithm–the only difference does not make a difference;
3. However, the transmission factor $\mathbf{T}(i)$ *does* make a difference in the multichannel problem. It appears both in (3.19) and as the matrix factor $(I - \mathbf{r}(i)^2)^{-1/2}$ in (3.20). It does not appear in the 2-D layer stripping algorithm (3.12), or in the algorithms of [4] and [5]. The reason is that since $\mathbf{r}(i)$ is scaled by Δ in algorithms obtained by discretizing continous equations, $(I - \mathbf{r}(i)^2)^{-1/2}$ is a term of *second order* in Δ, so that it will not be accounted for in first order discretizations.

3.7. Solution by Schur Algorithm and Block-Toeplitz Systems of Equation. The 2-D discrete inverse scattering problem defined by (3.14)-(3.16), or equivalently (3.20), can be solved very easily using the multichannel form of the Schur algorithm [18]. This algorithm has the same form as (3.12), except that (3.12a) should be replaced by (3.20) in order to incorporate the factor $(I - \mathbf{r}(i)^2)^{-1/2}$, which produces additional coupling between

channels. Unlike the scalar case, in which the Schur algorithm was identical to the layer stripping algorithm (2.9), the transmission factor does not cancel out, since it cannot be pulled out as an overall factor multiplying everything (matrices do not commute). As before, the independent variables must be scaled by Δ.

As before, the multichannel Schur algorithm should be expected to work well, since it solves exactly an explicitly discrete 2-D inverse scattering problem, instead of being merely a discretization of continuous equations. The discrete matrix two-component system (3.20) is lossless as long as the maximum singular value of $\mathbf{r}(i)$ is less than unity. The necessity of this should be clear from the factor $(I - \mathbf{r}(i)^2)^{-1/2}$; sufficiency can be established using a scattering argument, as before. This can often be established physically for media described by scattering systems of the form (3.20). For example, P and SV wave propagation in elastic media can be put into the form (3.20), where the matrices are all 2×2 [8]; the matrix of P-P, P-SV, and SV-SV interface reflection coefficients can be shown to have singular values less than unity (see the Appendix of [8]). This can be viewed as a matrix generalization of the scalar acoustic medium result $r(i) = (Z(i+1) - Z(i))/(Z(i+1) + Z(i)) \rightarrow |r(i)| < 1$.

Now suppose, as in the 1-D case, that the top interface is a free surface (perfect reflector), and that the probing impulse in (3.16) is introduced just below this surface. The Schur algorithm computes the reflection coefficients associated with the block-Toeplitz system of equations [18] (compare to (2.16))

$$\begin{bmatrix} I & \mathbf{R}(1) & \cdots & \mathbf{R}(n) \\ \mathbf{R}(1) & I & \cdots & \mathbf{R}(n-1) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{R}(n) & \mathbf{R}(n-1) & \cdots & I \end{bmatrix} \begin{bmatrix} \mathbf{F}_n^0 - \mathbf{G}_n^n \\ \vdots \\ \mathbf{F}_n^n - \mathbf{G}_n^0 \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \mathbf{T}(n) \end{bmatrix} \tag{3.21a}$$

$$\mathbf{R}(m)_{i,j} = R(m, i\Delta; j\Delta). \tag{3.21b}$$

Note that the $\mathbf{R}(m)$ are themselves matrices defined from the reflection response $R(m, k_x; k_x')$ defined in (3.16). $\mathbf{F}_n^i$ and $\mathbf{G}_n^i$ are elements of the matrix Green's function of (3.20). Reflection coefficients $\mathbf{r}(n)$ can be recovered using $\mathbf{r}(n) = \mathbf{G}_n^0 - \mathbf{F}_n^n$. All of these are generalizations of results in [10]; they appeared in [8] for the case of 2×2 matrices, which generalize directly to the $N \times N$ case.

In (3.21), scale all indices by Δ and let $\Delta \rightarrow 0$. The result is a 2-D generalization of the Krein integral equation. Solution of this integral equation will solve the original

continuous Schrodinger equation (3.1) inverse scattering problem, since the effects of all index shifts and transmission loss disappear in the continuous limit $\Delta \to 0$. Although we do not pursue this further here, this integral equation, which reconstructs a 2-D potential from backscattering data over a plane, seems to be new.

3.8. 2-D Generalization of the Kunetz Condition. Alternatively, (3.21) can be regarded [16] as a set of multichannel Yule-Walker equations for computing the least-squares multichannel autoregressive prediction filter coefficients for a discrete-time multichannel zero-mean stationary random process having matrix covariance lags $\mathbf{R}(i)$. The covariance matrix of the multichannel process at a given time is normalized to I. The transmission loss factor $\mathbf{T}(n)$ in (3.21) is analogous to the error covariance matrix of an n^{th}-order filter. This leads directly to the next result.

The matrix covariance lags of a multichannel stationary random process must form a positive definite sequence, i.e., the system matrix in (3.21a) must be positive definite. Since solution of the inverse scattering problem defined by (3.14)-(3.16) (or (3.20) and (3.16)) is equivalent to the solution of the block Toeplitz system (3.21a), we have the following result: *Let $R(n, k_x; k'_x)$ be the discrete-time Fourier transform (taking lateral position j into k_x) of the free-surface impulse response to excitation $\delta(n)\delta(k_x - k'_x)$ (see (3.16)) of a 2-D discrete scattering medium described by the 2-D discrete Schrodinger equation (3.13). Then $R(n, k_x; k'_x)$ must be a positive definite matrix function, i.e., the system matrix in (3.21a) must be positive definite.*

This result can be viewed as a 2-D generalization of the 1-D result of Kunetz [7]. The result of [7] was extended to the two-channel case (elastic medium) in [8]; the proof in [8], which follows closely the proof of the scalar result in [7], easily extends to the case of an arbitrary number of channels, and will not be repeated here. As in the 1-D case, the free-surface reflection response is in fact the autocorrelation of the transmission response $T(n, k_x; k'_x)$. Furthermore, a result from multichannel linear prediction theory [16] states that the system matrix in (3.21a) is positive definite if and only if the reflection coefficients $\mathbf{r}(i)$ have maximum singular values less than one.

This result carries the same significance as does the 1-D result. If this property does not hold, then the data are *infeasible*, in that no 2-D discrete medium described by (3.13)

could have produced it. Then some $\mathbf{r}(i)$ will have a singular value greater than one, and the multichannel Schur algorithm will fail. If the noisy data are processed so that it is positive definite, then the multichannel Schur algorithm will reconstruct the medium associated with the noisy data, not the actual medium, but it will not diverge.

3.9. Structure of Matrices r(i) and V(i). In this section we discuss the structure of $\mathbf{r}(i)$ and $\mathbf{V}(i)$, with regard to Hermitian symmetry and diagonal+Toeplitz or diagonal+circulant structure.

First, note the above results depend on the Hermitian symmetry of $\mathbf{r}(i)$, i.e., $\mathbf{r}(i) = \mathbf{r}(i)^H$. By reciprocity, we have that $R(n, k_x; k_x') = R(-n, -k_x'; -k_x)$, since the lateral wavenumber k_x specifies the direction of the plane wave $e^{-ikz}e^{-ik_x x}$, and reversing direction requires changing the sign of both k (hence the sign change of n) and k_x. Then each block in (3.21a) $\mathbf{R}(m)_{i,j} = R(m, i\Delta; j\Delta)$ (see (3.21b)) is Hermitian, since the sign change of n produces complex conjugation and the point $(-j, -i)$ is the transpose (about the line $i = -j$) of the point (i, j).

Alternatively, the coupled wave system (3.20), to be physically meaningful, must have the coupling between the i^{th} and j^{th} channels be the complex conjugate of the coupling between the j^{th} and i^{th} channels. This is a direct statement of Hermitian symmetry of reflection coefficients $\mathbf{r}(i)$.

For the continuous problem, the Hermitian symmetry of $\mathbf{r}(z)$ immediately implies, using (3.9), that the potential $\mathbf{V}(z)$ is also Hermitian (note that if $\mathbf{r}$ is Hermitian, then $\mathbf{r}^2 = \mathbf{r}\mathbf{r}^H$ is also). This is less apparent for the discrete problem; however, from (3.18) it is clear that $\mathbf{V}(i)$ is Hermitian, since $V(i, j)$ is real.

Second, a significant feature of $\mathbf{V}(i)$ for the discrete problem (see (3.18)) and $\mathbf{V}(z)$ for the continuous problem (see (3.8)) is that they are diagonal+Toeplitz or diagonal+circulant (depending on the discretization used). Since they are also Hermitian, this means they have only N degrees of freedom, not the N^2 degrees of freedom an arbitrary matrix would have (recall that the diagonal part is known). We have not yet exploited this structure.

Since $\mathbf{r}(z)$ is defined from $\mathbf{V}(z)$ using (3.9), and $\mathbf{r}(i)$ is defined from $\mathbf{V}(i)$ using (3.19a), it is clear that $\mathbf{r}(z)$ and $\mathbf{r}(i)$ also have only N degrees of freedom. This implies that (3.12b) in the 2-D layer stripping algorithm or multichannel Schur algorithm may be replaced by

solution of $\mathbf{rD} = \mathbf{U}$, where $\mathbf{D}$ and $\mathbf{U}$ are vectors, instead of matrices (since this constitutes N equations in N unknowns). The significance of this observation is that since different columns of $\mathbf{D}$ and $\mathbf{U}$ correspond to different experiments, i.e., different excitations, *probing at normal incidence only* is sufficient to reconstruct the medium.

This agrees with the requirements of the algorithm of [4] and [5], in which an asymmetric multichannel two-component wave system was used. The wave system in [4] and [5] had $\mathbf{V}(z)$ itself as one of the scatterers; since $\mathbf{V}(z)$ was circulant by the discretization used therein, $\mathbf{VD} = \mathbf{U}$ could be solved using the discrete Fourier transform. Unfortunately, the reflection coefficients in the symmetric two-component wave system used in this paper do not exhibit their structure in such an obvious manner. But normal incidence alone is sufficient in principle. This leads to the following result.

4. New 2-D Results for Point-Source Excitation.

4.1. Problem Formulation. The problem considered in this section is as follows. The medium is again described by the Schrodinger equation (3.1), with the same assumptions on the scattering potential. However, we now assume a free (perfectly reflecting) surface, and as excitation the 2-D impulsive point source ($1(\cdot)$ is the unit step)

$$\frac{1(t-\sqrt{x^2+z^2})}{2\pi\sqrt{t^2-(x^2+z^2)}} = \frac{1}{(2\pi)^3}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\frac{e^{ik_z z}}{-k_z^2}e^{ik_x x}e^{ikt}dk_x\,dk_z\,dk$$

$$= \mathcal{F}^{-1}_{k_x\to x}\mathcal{F}^{-1}_{k_z\to z}\mathcal{F}^{-1}_{k\to t}\{\frac{1}{-k_z^2}\},\quad k_z=\sqrt{k^2-k_x^2}. \tag{4.1}$$

The following comments are appropriate here:

1. Note that k_z is vertical wavenumber;
2. Note that the excitation (4.1) is actually a line source in 3-D, but since this paper deals exclusively with 1-D and 2-D, we call it a point source;
3. Note that (4.1) is the impulse response (Green's function) for a 2-D homogeneous medium. That is, it is the inverse temporal ($k \to t$) Fourier transform of the solution to the Schrodinger equation (3.1) if $\hat{V}(x,z) = 0$;
4. Note that (4.1) expresses the wave field in the time domain as a superposition of plane wave basis functions $e^{ik_z z}e^{ik_x x}e^{ikt}$.

Taking Fourier transforms in the lateral variable ($x \to k_x$) as in Section 3, the downgoing part of the excitation (4.1) at wavenumber k_x can be written as $\frac{e^{-ik_z z}}{-k_z^2}$, which can also be viewed as a downgoing solution to (3.3) when the potential $V(z, k_x) = 0$. The upgoing reflection response to the excitation (4.1) can then be written as $R(k_x, k_x)\frac{e^{ik_z z}}{-k_z^2}$. That is, the entire wave field $\hat{u}(x, z, k)$ in the Schrodinger equation (3.1) at the free surface $z = 0$ can then be written as the superposition

$$\hat{u}(x, z, k) = \int_{-\infty}^{\infty} \left(\frac{e^{-ik_z z}}{-k_z^2} + R(k_x, k_x)\frac{e^{-ik_z z}}{-k_z^2} + R(k_x, k)\frac{e^{ik_z z}}{-k_z^2} \right) e^{ik_x x} dk_x \qquad (4.2)$$

of a downgoing incident field due to the source and reflection off of the free surface, and an upgoing scattered field measured at the surface $z = 0$. The goal is to reconstruct potential $\hat{V}(x, z)$ from reflection response $R(k_x, k_x)$.

4.2. New Results. Formally changing variables from k to k_z, (3.3) becomes

$$\left(\frac{d^2}{dz^2} + k_z^2 \right) u(z, k_x, k_x) = \int_{-\infty}^{\infty} V(z, k_x - k_x')u(z, k_x, k_x')dk_x', \qquad (3.3)$$

where $u(z, k_x, k_x) = u(z, k = \sqrt{k_x^2 + k_z^2}, k_x)$ in (3.4) (we do not bother to introduce new notation for $u(z, k_x, k_x)$).

From this point forward, we may proceed exactly as in Section 3, with (4.3) replacing (3.3). There is one major change: The $-k_x^2$ term in (3.3) has disappeared, absorbed into k_z^2 in (4.3). The effect of this change is that $F(k_x^2)$ defined in (3.5) and used subsequently in (3.6) and (3.7) is now zero. This means that $\mathbf{V}_{(i,j)}$ defined in (3.8) no longer has diagonal+Toeplitz or diagonal+circulant structure (depending on discretization; see Section 3.4), but has purely Toeplitz or circulant structure.

This produces an important simplification in the results of Section 3. Suppose that the discretization used produces circulant structure in $\mathbf{V}_{(i,j)}$. Then the reflection coefficients $\mathbf{r}_{(i,j)}$ defined in (3.9) will also be circulant. This implies that the discretized reflection response matrices $\mathbf{R}(k_x)_{(i,j)} = R(k_x, k_x = (i - j)\Delta)$ are circulant, since the $\mathbf{R}(k_x)$ can be generated from the $\mathbf{r}$ by running the 2-D layer stripping algorithm (3.12) backwards. Finally, this shows that the block-Toeplitz system of equations (3.21) is now block-Toeplitz with circulant blocks, so it can be assembled from $R(k_x, k_x)$, even though these have fewer degrees of freedom than the $R(k, k_x; k_x')$ used before.

The 2-D layer stripping algorithm can then be used to reconstruct the circulant reflection coefficient matrices from the circulant reflection response matrices. This is an improvement over the result in Section 3, since less data are needed to reconstruct the potential. The reason for this is that the structure noted in Section 3.9 is brought out by taking the superposition of plane-wave impulse reflection responses (4.2), rather than considering each response separately. The simplification appears in the multichannel Schur algorithm as the simplified form of this algorithm as applied to matrices with Toeplitz-block-Toeplitz, rather than merely block-Toeplitz, structure.

Now suppose the equations derived for the discrete inverse scattering problem are used. Note that the more complicated equations (3.19) relating the discrete potential to the reflection coefficients still preserve the circulant structure, as circulant matrices are closed under addition, multiplication, and inversion. This also leads to the following second 2-D generalization of the Kunetz result. *Let $R(k_x, k_x)$ be the Fourier transform of the free-surface response of a 2-D scattering medium described by the 2-D Schrodinger equation (3.1), to a point-source excitation, as specified in (4.1)-(4.2). Then $R(k_x, k_x)$ must be a 2-D positive definite function, i.e., the system matrix in (3.21a) must be positive definite.* Note that this differs from the result of Section 3, which specified a multichannel (vector) positive definite function–now a truly 2-D positive definite funciton is specified.

5. Numerical Examples. In this section we present some illustrative numerical examples. First, we demonstrate the algorithms, and show that they provide results superior to results using the Born approximation. Next, we add noise to the data to render it infeasible, and show that the algorithms fail, as expected. Finally, we project the noisy data onto the space of positive definite discrete functions, and show that the algorithms no longer diverge.

6. Conclusion.

Acknowledgment. This research was supported by the Office of Naval Research under grant #N00014-90-J-1897.

References.

1. S. COEN, M. CHENEY, AND A. WEGLEIN, *Velocity and Density of a Two Dimensional Acoustic Medium from Point Source Surface Data*, J. Math Phys., 25 (1984), pp. 1857-1860.
2. A.E. YAGLE, *A Layer Stripping Fast Algorithm for the Two-Dimensional Direct Current Inverse Resistivity Problem*, IEEE Trans. Geosci. and Rem. Sensing, GE-25 (1987), pp. 558-563.
3. M. CHENEY, *Inverse Scattering in Dimension Two*, J. Math Phys., 25 (1984), pp. 94-102.
4. A.E. YAGLE AND B.C. LEVY, *Layer Stripping Solutions of Multidimensional Inverse Scattering Problems*, J. Math Phys., 27 (1986), pp. 1701-1710.
5. A.E. YAGLE AND P. RAADHAKRISHNAN, *Numerical Performance of Layer Stripping Algorithms for Two-Dimensional Inverse Scattering Problems*, Inverse Problems, 8 (1992), pp. 645-665.
6. V. WESTON, *Invariant Imbedding for the Wave Equation in Three Dimensions and the Applications to the Direct and Inverse Problems*, Inverse Problems, 6 (1990), pp. 1075-1105.
7. G. KUNETZ AND I. d'ERCEVILLE, *Sur Certaine Proprieties d'une Onde Acoustique Plane de Compression dans un Milieu Stratifie*, Annales de Geophysique, 18 (1962), pp. 351-359.
8. C.W. FRASIER, *Discrete-Time Solution of Plane P-SV Waves in a Plane Layered Medium*, Geophysics, 35 (1970), pp. 197-219.
9. A.M. BRUCKSTEIN, B.C. LEVY, AND T. KAILATH, *Differential Methods in Inverse Scattering*, SIAM J. Appl. Math, 45 (1985), pp. 312-335.
10. A.M. BRUCKSTEIN AND T. KAILATH, *Inverse Scattering for Discrete Transmission-Line Models*, SIAM Review, 29 (1987), pp. 359-389.
11. J. WARE AND K. AKI, *Continuous and Discrete Inverse Scattering Problems in a Stratified Elastic Medium. Part I*, J. Acoust. Soc. Am., 45 (1969), pp. 911-921.
12. K.M. CASE AND M. KAC, *A Discrete Version of the Inverse Scattering Problem*, J. Math. Phys., 14 (1973), pp. 594-603.
13. C.B. SHARPE, *An Inverse Scattering Problem Associated with the Discrete Schrodinger*

Equation, SIAM J. Appl. Math., 32 (1977), pp. 405-417.

14. P. DELSARTE AND Y. GENIN, *The Split Levinson Algorithm*, IEEE Trans. Acoust., Speech, and Sig. Proc., ASSP-34 (1986), pp. 470-478.
15. A.E. YAGLE, *Fast Algorithms for Estimation and Signal Processing: An Inverse Scattering Framework*, IEEE Trans. Acoust., Speech, Sig. Proc., 37 (1989), pp. 957-959.
16. S.M. KAY, *Modern Spectral Estimation*, Prentice-Hall (1988).
17. A.E. YAGLE, *Discrete Matrix Schrodinger Equation Equivalents of Discrete Two-Component Matrix Wave Systems*, Inverse Problems, 5 (1989), pp. 425-436.
18. J. RISSANEN, *Algorithms for Triangular Decomposition of Block Hankel and Toeplitz Matrices with Applications to Factoring Positive Matrix Polynomials*, Math. of Comput., 27 (1973), pp. 147-154.

APPENDIX C

A.E. Yagle, "Multiresolution Algorithms for Solving One-Dimensional Inverse Scattering Problems Using the Wavelet Transform," revision submitted to IEEE Trans. Sig. Proc., October 1993.

This paper applies the wavelet transform to the 1-D inverse scattering problem. We derive a layer stripping algorithm that uses as input the wavelet transform of the impulse reflection response of the medium. The wavelet transform decouples the layer stripping algorithm into a set of wave systems at differing wave speeds. Any of these multiple-resolution systems could be used to reconstruct the medium; this allows some flexibility on how the data are used.

We also derive both a layer stripping algorithm, and a linear system of equations, in the 2-D (time and space) wavelet transform domain, from the layer stripping algorithm and Krein integral equation, respectively. These results show how data at one resolution affects the reconstruction at another resolution. They are interpreted in terms of fast algorithms for the slanted Toeplitz structured linear system of equations. The Born approximation is also derived and discussed.

Multiresolution Algorithms for Solving One-Dimensional Inverse Scattering Problems Using the Wavelet Transform

Andrew E. Yagle
Department of Electrical Engineering and Computer Science
The University of Michigan, Ann Arbor, MI 48109-2122
July 1992

Abstract

The wavelet transform is applied to the inverse scattering problem of reconstructing the reflectivity function of a two-component wave system from its impulse reflection response, a problem which has applications in acoustics and some synthesis problems. Three new multiresolution algorithms are obtained. First, a single wavelet transform in time results in a set of independent wave systems in which waves propagate at a speed determined by the dilation factor. Second, wavelet transforms in both time and space result in a set of coupled wave systems; in each system the waves propagate at a speed determined by the two dilation factors. Finally, wavelet transforms in both time and space are applied to the Krein integral equation, resulting in a block-slanted-Toeplitz linear system of equations, to which the coupled wave system is related. The latter two results relate the wavelet transform of the reflectivity function to the wavelet transform of the impulse reflection response.

EDICS number: 4.4. Phone number: 313-763-9810

I. INTRODUCTION

The mathematical inverse problem of reconstructing a one-dimensional continuous layered medium from its impulse reflection response has many applications in many different fields. These include reflection seismology [1], acoustic measurement of the shape of the human vocal tract [2], and the synthesis of nonuniform transmission lines [3]. All of these problems can be formulated as a nonlinear problem of reconstructing a spatially-varying reflectivity function, in the two-component wave system (1) below, from a temporally-varying impulse reflection response function. Note the problem is nonlinear due to multiple scattering in the wave system; these effects are included throughout this paper, unlike some methods which ignore multiple scattering (the Born approximation).

There are two basic approaches to solving this inverse problem. The first approach is to solve the Krein integral equation [4], which has the impulse reflection response as its kernel, and read off the reflectivity function from the solution. The second approach is to use a layer-stripping algorithm [5],[6], which operates recursively in space by differentially reconstructing the reflectivity function, and then differentially propagating the waves in (1). Layer stripping algorithms are computationally more efficient than solving integral equations. The relation between integral equations and layer stripping algorithms is a continuous-parameter analog of the relation between Toeplitz systems of equations and the Levinson and Schur fast algorithms for solving such systems of equations; see [6] for a detailed treatment.

The wavelet transform has recently received much attention from the signal processing community [7],[8]. The wavelet transform is a multiresolution decomposition of a function in terms of scalings (contractions and dilations) and translations of a wavelet basis function, which is localized in time and frequency. It thus generates a time-frequency (actually time-scale) representation of the function. The wavelet transform can be related to multirate filtering, subband coding, and quadrature-mirror filtering. Two important papers on wavelets are [9] and [10]; the list of references in [7] gives some idea of the enormous amount of interest and applications the wavelet transform has recently generated. No attempt will be made here to summarize all of the recent work on wavelets.

This paper applies the wavelet transform to the inverse problem of reconstructing the reflectivity function of a two-component wave system (1) from its impulse reflection response. Since this inverse problem is nonlinear, the resulting multiresolution algorithms are complicated; however, they have interesting interpretations, both physical and in terms of fast algorithms. The significance of these results is as follows: (1) the algorithms provide a space-time-frequency (Section III) or space-wavenumber-time-frequency (Sections IV and V) representation of the inverse problem solution process; (2) the algorithm of Section III operates even in the presence of time-and-frequency support-limited noise or interference, unlike previous approaches; (3) the algorithms of Sections IV and V show how the data at one resolution is coupled to the solution at another resolution; and (4) the new coupled wave systems are related to multichannel coupled lattice structures associated with the new block-slant-Toeplitz structured linear system of equations derived in Section V. New contributions of this paper include all of the results of Sections III-V.

This paper is organized as follows. In Section II we quickly review the two-component wave system inverse scattering problem, along with some of its applications, the layer stripping algorithm for solving this problem, and the Krein integral equation for solving this problem. We also quickly review the discrete wavelet expansion of a continuous function in terms of wavelet basis functions which are orthonormal at different scales and translations. In Section III the wavelet transform with respect to time is applied to the layer stripping algorithm, resulting in a set of independent wave systems in which the waves propagate at a speed determined by the dilation factor. This allows a different wave system to be used to compute the reflectivity function at each depth, reducing the effects of noise in the impulse reflection response.

In Section IV wavelet transforms with respect to both time and space are applied to the layer stripping algorithm. This results in a set of coupled wave systems; in each system the waves propagate at a speed determined by the two dilation factors. This allows the reflectivity function to be reconstructed at different resolutions, from the impulse reflection response at different resolutions (although not independently). We also specialize to the specific results when a Haar wavelet basis function is used, and interpret the resulting algorithm in the Born approximation. In Section V wavelet transforms with respect to

both time and space are applied to the Krein integral equation. This results in a block-structured linear system of equations in which each block has a slanted-Toeplitz structure, viz. the elements of a given block are constant along diagonals with slope determined by the location of the block. This structure is related to the form of the multichannel coupled wave system derived in Section IV. Section VI concludes the paper with a summary.

II. QUICK REVIEW OF INVERSE SCATTERING AND THE WAVELET TRANSFORM

A. The 1-D Inverse Scattering Problem

The basic one-dimensional inverse scattering problem considered in this paper is as follows. Let x be a spatial variable, which we will call depth (increasing downward), and t be time. A scattering medium is described by the *two-component wave system*

$$\frac{\partial}{\partial x}\begin{bmatrix} d(x,t) \\ u(x,t) \end{bmatrix} = \begin{bmatrix} -\frac{\partial}{\partial t} & -r(x) \\ -r(x) & \frac{\partial}{\partial t} \end{bmatrix} \begin{bmatrix} d(x,t) \\ u(x,t) \end{bmatrix}, \quad 0 < x < 1 \tag{1}$$

where the *reflectivity function* $r(x)$ characterizes the scattering medium. The scattering medium is thus a chain or series of differential scattering sections, each having the form shown in Figure 1 (note the lattice structure of the medium, similar to the lattice filters of linear prediction).

Note that if $r(x) = 0$, then $d(x,t) = d(x-t)$ and $u(x,t) = u(x+t)$. Thus $d(x,t)$ and $u(x,t)$ can be interpreted as Downgoing and Upgoing waves, scattered into each other at depth x by the reflectivity function $r(x)$. The scattering medium is assumed to have finite extent in x; without loss of generality this extent is scaled to $0 < x < 1$. The boundary conditions are a radiation condition at $x = 1$ ($d(x,t) = d(x-t)$ and $u(x,t) = 0$ for $x > 1$) and a free surface at $x = 0$ ($d(0,t) = u(0,t)$, excluding sources). The free surface implies that an upgoing wave at $x = 0$ is simply reflected into a downgoing wave.

This scattering medium is probed with an impulsive plane wave $\delta(t-x)$, which propagates downward into the medium in increasing depth x as time t increases. The reflection response $k(t)$ of the medium to this impulse is measured at $x = 0$. This amounts to initializing (1) with

$$d(0,t) = \delta(t) + k(t); \quad u(0,t) = k(t). \tag{2}$$

The inverse scattering problem is then to compute the reflectivity function $r(x)$ from the impulse reflection response $k(t)$.

Several types of inverse problems can be formulated as the above problem [1]-[3]. For example, if the scattering medium is a continuously-layered acoustic medium with constant wave speed and varying density $\rho(x)$, then the problem of reconstructing $\rho(x)$ from the reflection response of the medium to an impulsive plane wave $\delta(t-x)$ can be formulated as (1) by defining [5]

$$d(x,t) = \frac{p(x,t)}{\sqrt{\rho(x)}} + \sqrt{\rho(x)}v(x,t); \quad u(x,t) = \frac{p(x,t)}{\sqrt{\rho(x)}} - \sqrt{\rho(x)}v(x,t); \quad r(x) = \frac{1}{2}\frac{d\rho}{dx}, \quad (3)$$

where $p(x,t)$ is pressure in the medium and $v(x,t)$ is velocity of the medium. In other applications $\rho(x)$ is replaced with local impedance of a nonuniform transmission line [2] or cross-sectional area of the human vocal tract [3]. See the references in [6] for more details.

B. Layer Stripping Solution to the 1-D Inverse Scattering Problem

It is clear from the forms of (1) and (2) that $d(x,t)$ and $u(x,t)$ are causal functions, i.e., $d(x,t) = u(x,t) = 0$ for $t < x$. Discretizing depth x and time t into integer multiples of a small constant Δ and using forward differences to approximate the partial derivatives in (1), the two-component wave system (1) discretizes into [5],[6]

$$\begin{bmatrix} d(x+\Delta, t+\Delta) \\ u(x+\Delta, t-\Delta) \end{bmatrix} = \begin{bmatrix} 1 & -r(x)\Delta \\ -r(x)\Delta & 1 \end{bmatrix} \begin{bmatrix} d(x,t) \\ u(x,t) \end{bmatrix} \quad (4a)$$

$$r(x)\Delta = u(x,x)/d(x,x) = u(x,x) \Big/ \prod_{i=0}^{x/\Delta - 1} (1 - r(i\Delta)^2\Delta^2). \quad (4b)$$

Equation (4b) follows from setting $t = x$ in (4a) and noting that $u(x+\Delta, x-\Delta) = 0$ by causality. For $\Delta << 1$ we have $d(x,x) \approx 1$.

Equations (4), initialized using $d(0,t) = u(0,t) = k(t)$, can be recursively propagated in increasing depth x, recovering the reflectivity function $r(x)$ along the way. This is the essential idea of a ***layer stripping algorithm.*** Recall that by causality $d(x,t) = u(x,t) = 0$ for $t < x$, so (4a) is only propagated for $t \geq x$; at $t = x$ the second equation of (4a) is equivalent to (4b). Also note that $d(x = m\Delta, t = n\Delta) = u(x = m\Delta, t = n\Delta) = 0$ if $m + n$

is odd. The nonlinear multiple reflections are taken into account by the nonlinear product terms in (4a). Such algorithms have been derived in many forms for many problems [1],[6].

We recognize (4) as the *Schur algorithm* of signal processing. This algorithm has a long and rich history dating back to 1917; see [11]. Note that (4) has the nice physical interpretation of reconstructing a discrete layered medium, consisting of homogeneous layers of thickness Δ. It arises here because such a discrete layered medium can also be reconstructed by solving a Toeplitz or Hankel system of equations; the Schur algorithm here is simply a fast algorithm to obtain the $r(x)\Delta$ [12].

C. Integral Equation Solution to the 1-D Inverse Scattering Problem

An alternative to the layer stripping algorithm (4) is to solve the *Krein integral equation* [4]

$$k(x-t) = h(x,t) + \int_{-x}^{x} h(x,z)k(|z-t|)dz; \quad |t| \le x; \quad 0 \le x \le 1 \tag{5}$$

for $h(x,t)$. $r(x)$ can then be computed from $h(x,t)$ using

$$r(x) = 2h(x,-x). \tag{6}$$

Note the Toeplitz structure in the kernel $k(|z-t|)$ of (5). This structure shows why, from a fast algorithms perspective, the Schur algorithm (4) can be used to reconstruct $r(x)$ from $k(t)$ more quickly than solving the integral equation (5) by discretizing it into a Toeplitz system of equations.

We recognize (5) as the Wiener-Hopf integral equation for computing the linear least-squares filter $h(x,t)$ for estimating a zero-mean wide-sense stationary random process, with covariance function $k(|x-t|)$, at time x from noisy observations, with additive white noise, measured over the interval $-x < t < x$. Equation (6) then merely states the well-known result of linear prediction that the reflectivity function (continuous reflection coefficient) equals the filter weight at the far end of the interval of observation. This illustrates the well-known connection between linear prediction and inverse scattering.

Applying the operator $\left(\frac{\partial}{\partial x} + \frac{\partial}{\partial t}\right)$ to (5) and using linearity and uniqueness of the

solution of (5) results in the *Krein-Levinson equation* [13]

$$\left(\frac{\partial}{\partial x}+\frac{\partial}{\partial t}\right)h(x,t)=-r(x)h(x,-t); \quad |t|<x \tag{7a}$$

$$r(x)=h(x,-x)=k(2x)-\int_{-x}^{x}h(x,z)k(x+z)dz. \tag{7b}$$

Discretizing (7) in the same way that (1) was discretized to (4a) results in the familiar Levinson algorithm for solving the discretized integral equation (6). Note that (7b) discretizes into the "inner product" computation of the Levinson algorithm. See [6] and [12] for more details. Note also that (7a) has the same form as the first equation of (1). This illustrates that the form of the lattice equations (1) is related to any solution procedure for the inverse scattering problem; see [6] and [12].

We note in passing that the two-component wave system (1) inverse scattering problem can be transformed into a Schrodinger equation inverse scattering problem, and vice-versa, provided the functions are all smooth enough. The relations between the well-known Schrodinger equation inverse scattering machinery, such as the Gel'fand-Levitan and Marchenko integral equations, and the Krein integral equation and layer stripping algorithm, are discussed in [6]. One important advantage of the two-component wave system (1) over the Schrodinger equation is that double differentiations are not required.

D. Discrete Orthonormal Wavelet Transforms of Continuous Functions

The discrete orthonormal ***wavelet transform*** or representation $F(m,n)$ of a continuous square-integrable function $f(x)$ is [9],[10]

$$F(m,n)=\int_{-\infty}^{\infty}f(x)2^{m/2}\phi(2^m x-n)dx \tag{8a}$$

$$f(x)=\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty}F(m,n)2^{m/2}\phi(2^m x-n) \tag{8b}$$

where $\phi(x)$ is the ***wavelet basis function.*** $\phi(x)$ has the properties that it is orthogonal (in the sense of the usual L^2 inner product) to its ***scalings*** $\phi(2^m x)$ (dilations for $m<0$; compressions for $m>0$) and to the ***translations*** $\phi(2^m x-n)$ of its scalings, and the set of all

scalings and translations $\{2^{m/2}\phi(2^m x - n); m, n \in integers\}$ forms a complete orthonormal set (the factor $2^{m/2}$ is necessary for ortho-*normality*). Algorithms for obtaining suitable basis functions $\phi(t)$ are discussed in [9] and [14].

The significance of the wavelet transform or expansion (8b) is as follows. Coarse scalings (small m) amount to convolving $f(x)$ with a basis function that is drawn out in time, and then sampling. Equivalently, $f(x)$ itself can be viewed as being compressed, and then filtered with $\phi(x)$. Either way, there is clearly a loss of resolution–only long-scale features of $f(x)$ are captured in $F(m,n)$ when m is small. On the other hand, fine scalings (large m) amount to convolving $f(x)$ with a tightly compacted basis function, and then sampling, so that the resolution is quite high. Furthermore, since the basis function is so tightly compacted, the high resolution is local, in that the wavelet transform at a given point depends only on $f(x)$ in the immediate vicinity of that point. Thus the fine scale information is localized, while the coarse scale information is global.

Because the coarse scale wavelet transform carries only low-frequency (long-scale) information, it may be sampled coarsely, with long intervals between the sample points. The fine scale wavelet transform carries high-resolution (small-scale) information, and so it must be sampled finely. It can be shown that the sampling grid shown in Figure 2 is sufficient to completely represent $f(x)$, provided $\phi(x)$ is chosen properly [14]. Furthermore, each fine-scale sample will carry localized information about the local fine-scale behavior of $f(x)$. If the basis function $\phi(x)$ is localized in both time and frequency, the wavelet transform will be a type of time-frequency representation of $f(x)$. In [7] wavelet analysis is noted as being analogous to a microscope–at low magnifications, most of the object is visible, while at high magnifications, only a small part of the object is visible, and many translations of it are needed to view all of the object.

The simplest example of a wavelet transform is to choose $\phi(x)$ as the Haar basis function, shown in Figure 3. The wavelet transform (8a) of $f(x)$ is simply the Haar transform or expansion of $f(x)$. The algorithm of Section IV will be specialized to this particular expansion; other choices of wavelet basis function $\phi(x)$ are also possible. In general, the wavelet transform of $f(x)$ can be computed quickly using filter banks. Finally, we mention in passing that there are many types of wavelet transforms; in this paper, our

attention is restricted to (8).

III. WAVELET TRANSFORM IN TIME ONLY

A. Derivation of Wavelet Layer Stripping Algorithm

In this section we show how $r(x)$ can be differentially reconstructed from the wavelet transform of $k(x)$. The wavelet transform of (4) results in an uncoupled set of discrete wave systems, each of which may be propagated separately. This results in some flexibility on how layer stripping is used.

Let $\phi(x)$ be any wavelet basis function with compact support on the interval $0 < x < 1$ (in this section $\phi(x)$ need not be an orthonormal basis function). Apply the wavelet transform (8a), with x replaced by t, to equations (4). Note that now t is NOT discretized to integer multiples of Δ; it is a continuous independent variable. However x is still discretized. This yields the ***single wavelet transform (in time only) layer stripping algorithm***

$$\begin{bmatrix} D(x+\Delta; m, n+2^m\Delta) \\ U(x+\Delta; m, n-2^m\Delta) \end{bmatrix} = \begin{bmatrix} 1 & -r(x)\Delta \\ -r(x)\Delta & 1 \end{bmatrix} \begin{bmatrix} D(x; m, n) \\ U(x; m, n) \end{bmatrix} \tag{9a}$$

$$r(x)\Delta = U(x; m, n = 2^m x - 1)/D(x; m, n = 2^m x - 1). \tag{9b}$$

where $D(x; m, n)$ and $U(x; m, n)$ are the single wavelet transforms (in time only) of $d(x, t)$ and $u(x, t)$, respectively.

Equation (9b) is derived as follows. Recall that the wavelet basis function $\phi(x)$ has compact support on the interval $0 < x < 1$, and that by causality $d(x, t) = u(x, t) = 0$ for $t < x$. These imply that $D(x; m, n) = U(x; m, n) = 0$ for $n < 2^m x - 1$ (the -1 comes from the length of the support of $\phi(x)$). Setting $n = 2^m x - 1 < 2^m(x + \Delta) - 1$ in the second equation of (9a) then yields (9b).

Equation (9) is initialized using $D(0; m, n) = U(0; m, n) = K(m, n)$, where $K(m, n)$ is the wavelet transform of the impulse reflection response $k(t)$. Like (4), equation (9) can be recursively propagated in increasing depth x, recovering the reflectivity function $r(x)$ along the way. Since $D(x; m, n) = U(x; m, n) = 0$ for $n < 2^m x - 1$, equation (9) is only propagated for $n \geq 2^m x - 1$, so the size of the shift in the translation variable n is

appropriate.

B. Comments

Although equations (9) look similar to (4), there are several important distinctions between the two algorithms. We summarize those distinctions here:

1. When (4) is actually implemented on a computer as the Schur algorithm, x and t are discretized to integer multiples of Δ. In deriving (9) from (4), t is treated as a *continuous independent variable*, so that the wavelet transform with respect to t of (4) can be taken;
2. The wavelet transform breaks up the continuous (in t) equations (4) into a set of completely decoupled discrete-space discrete-time wave systems. The wave speed of the wave system at resolution m is 2^{-m}. This explains why $D(x;m,n)$ and $U(x;m,n)$ are zero for $n < 2^m x - 1$, while $d(x,t)$ and $u(x,t)$ are zero only for $t < x$–the finer the scale, the slower the wave speed;
3. Note that fractional values of the translation index n are required for scales m such that $2^m\Delta$ is not an integer; it is suggested that $\Delta = 2^{-M}$ for some integer M. This does not create any difficulties; to interpret this, see Figure 2. Fractional values of the translation index n simply correspond to samples taken in between the sample points of Figure 2; in essence, we have oversampled, redundant wavelet representations of $d(x,t)$ and $u(x,t)$, for each x;
4. Note that save for the wave speed of $2^{-m} \neq 1$, equations (9), for each m, have the same form as the Schur algorithm. The wavelet transform decouples the original layer stripping algorithm into a set of Schur-like algorithms. However, the original layer stripping algorithm requires that $k(t)$ be discretized in t to create a fully discrete algorithm, while the wavelet layer stripping algorithm uses the continuous function $k(t)$, expanded into a set of discrete time-frequency coefficients, each of which are then propagated in Schur-like algorithms;
5. As a result, the Schur-like recursions (10) can be interpreted as solving a set of multiresolution discrete inverse scattering problems. They can be replaced by discrete

Riccati equations, interpreted as fast algorithms for solving discretized integral equations, viewed as representing discrete transmission lines, etc. [12].

C. Implementations of Wavelet Layer Stripping Algorithm

The most significant aspect of (9) is the flexibility afforded by the complete decoupling of the discrete wave systems. In order to reconstruct $r(x)$ from the wavelet expansion $K(m,n)$ of the impulse reflection response $k(x)$, we may do any of the following:

1. Select any single scale m, and reconstruct $r(x)$ using solely the wave system at that scale. This requires $K(m,n)$ for a single m; fractional values of n will be required if $2^m\Delta$ is not an integer. This is not a problem, since we are given $k(x)$ as data, so we can compute its wavelet transform for any m and n.

This can be understood as follows. At coarse scales, the coarse sampling will not furnish sufficient information to reconstruct $r(x)$ discretized to integer multiples of Δ. At finer scales, the fine sampling furnishes more that sufficient information, and some samples are skipped–if $\Delta = 2^{-M}$ and $m > M$, only every $2^{(m-M)}$ sample (in n) of $K(m,n)$ will be used.

This choice should be made if interference or noise in the data $K(m,n)$ is present on most scales, but small or absent on a specific scale. Note that the wavelet basis function can be selected to bring about this situation, if possible.

2. We may start off using a single scale m_1, and then at depth $x = x_s$ switch to a coarser scale $m_2 < m_1$. This can be accomplished by subsampling the finer scale wavelet coefficients to obtain the coarser scale coefficients [9]. Since we are given $k(x)$, we know $K(m_1,n)$, and thus $D(x_s;m_1,n)$ and $U(x_s;m_1,n)$, for all n.

This choice should be made if interference or noise in the data is localized in the wavelet transform domain, i.e., present at scale m_1 and absent at scale m_2 for some translations, and then absent at scale m_1 and present at scale m_2 for further translations.

3. We can propagate all of the scales, and compute $r(x)$ at a given depth x by performing a least-squares fit to (9b), analogous to what was done in [5]. That is, we determine

the value of $r(x)$ that minimizes

$$\sum_m [r(x)\Delta - U(x; m, n = 2^m x - 1)/D(x; m, n = 2^m x - 1)]^2;$$

this of course is simply the mean of the $U(x; m, n = 2^m x - 1)/D(x; m, n = 2^m x - 1)$.

This choice should be made if interference or noise in the data is present at all scales. Since interference will have different effects on different scales, the various $r(x)$ computed from discrete wave systems at different scales m will differ; using all of the scales will result in an improved estimate of $r(x)$, as in [5].

D. Comparison Between Wavelet and Frequency Domain Implementations

It may seem surprising that a wavelet expansion of $k(t)$ can furnish different information about $r(x)$ at different scales–we seem to be exploiting redundant information, so that there should be no difference in results at different scales. To understand how this can happen, take the temporal Fourier transform of (1). This yields the differential Schur recursions [6]

$$\frac{d}{dx}\begin{bmatrix}\hat{d}(x,\omega)\\ \hat{u}(x,\omega)\end{bmatrix} = \begin{bmatrix}-j\omega & -r(x)\\ -r(x) & j\omega\end{bmatrix}\begin{bmatrix}\hat{d}(x,\omega)\\ \hat{u}(x,\omega)\end{bmatrix} \tag{10}$$

where $\hat{d}(x,\omega) = \int_x^\infty d(x,t)e^{-j\omega t}dt$, and similarly for $\hat{u}(x,\omega)$.

Now suppose (10) is initialized at the surface $x = 0$ with $\hat{d}(x,\omega) = \hat{u}(x,\omega) = \hat{k}(\omega)$. The reflectivity function $r(x)$ can then be computed from $\hat{u}(x,\omega)$ using either of two expressions [6]:

$$r(x) = \frac{1}{\pi}\int_{-\infty}^{\infty}\hat{u}(x,\omega)e^{j\omega x}d\omega = \lim_{\omega\to\infty} 2j\omega e^{j\omega x}\hat{u}(x,\omega). \tag{11}$$

The significance of (10) is that each spectral component is propagated separately–there is no mixing of components at different frequencies ω. This is not surprising–viewed as functions of time alone, (1) are linear (there are no products of two functions of t). This is analogous to the temporal wavelet transform of (4) resulting in a decoupled set of discrete wave systems, in which each scale or resolution propagates separately.

The significance of (11) is that once the equations have been propagated to depth x $r(x)$ may be reconstructed either from all of the components, or from *just the highest*

frequency components. That is, the highest-frequency components of (10) propagate completely independently of other components, since the second equality of (11) may be used to determine $r(x)$ independently of the other components. Thus, even if there is bandlimited noise in the data, $r(x)$ can still (in principle) be reconstructed without error–simply propagate only the highest-frequency components.

Something analogous to (10) and (11) is happening in (9a) and (9b). The differences are as follows. First, since the wavelet transform is redundant, each scale can be propagated independently of other scales, rather than just the highest-frequency components. Second, and more important, the wavelet transform allows use of representations other than those in terms of sinusoids. By choosing a proper wavelet basis function, interference can be confined to only some scales, or features of $k(t)$ can be efficiently represented, etc.

IV. WAVELET TRANSFORM IN SPACE AND TIME–LAYER STRIPPING

A. Derivation of Coupled Discrete Wave Systems

In this section we extend the previous result to the problem of reconstructing the wavelet transform of $r(x)$ directly from the wavelet transform of $k(x)$. That is, we explore the relation between $r(x)$ at different scales and $k(x)$ at different scales. Since the problem is nonlinear in variable x, due to the $r(x)u(x,t)$ and $r(x)d(x,t)$ terms, we expect this relation to be complicated, and it turns out to be so. The result is again a set of discrete wave systems, which are now coupled. However, an approximate algorithm is still possible.

Let $\phi(x)$ be an orthonormal wavelet basis function orthogonal to its scalings and translations (alternatively, $\phi(x)$ may have "tight frame" [14]) such that the set of its scalings and translations form a complete set, and having compact support on the interval $0 < x < 1$. Obviously the Haar basis function would be suitable here. Define the following two-dimensional wavelet expansions:

$$D(m_1, n_1, m_2, n_2) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} d(x,t) 2^{m_1/2} \phi(2^{m_1} x - n_1) 2^{m_2/2} \phi(2^{m_2} t - n_2)\, dx\, dt;$$

$$d(x,t) = \sum_{m_1} \sum_{n_1} \sum_{m_2} \sum_{n_2} D(m_1, n_1, m_2, n_2) 2^{m_1/2} \phi(2^{m_1} x - n_1) 2^{m_2/2} \phi(2^{m_2} t - n_2) \quad (12a)$$

$$U(m_1,n_1,m_2,n_2) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} u(x,t)2^{m_1/2}\phi(2^{m_1}x-n_1)2^{m_2/2}\phi(2^{m_2}t-n_2)\,dx\,dt;$$

$$u(x,t) = \sum_{m_1}\sum_{n_1}\sum_{m_2}\sum_{n_2} U(m_1,n_1,m_2,n_2)2^{m_1/2}\phi(2^{m_1}x-n_1)2^{m_2/2}\phi(2^{m_2}t-n_2) \quad (12b)$$

$$R(m_3,n_3) = \int_{-\infty}^{\infty} r(x)2^{m_3/2}\phi(2^{m_3}x-n_3)\,dx; \quad r(x) = \sum_{m_3}\sum_{n_3} R(m_3,n_3)2^{m_3/2}\phi(2^{m_3}x-n_3). \quad (12c)$$

We use odd-subscripted variables m_1, m_3, m_5, etc. to denote spatial variables, and even subscripted variables m_2, m_4, m_6, etc. to denote temporal variables, when spatial and temporal variables are both present.

Also define the discrete function

$$C(m_1,n_1,m_2,n_2,m_3,n_3) = \int_{-\infty}^{\infty} 2^{m_1/2}\phi(2^{m_1}x-n_1)2^{m_2/2}\phi(2^{m_2}x-n_2)2^{m_3/2}\phi(2^{m_3}x-n_3)\,dx. \quad (13)$$

It is clear that $C(m_1,n_1,m_2,n_2,m_3,n_3)$ is entirely symmetric with respect to permutations of the subscripts $\{1,2,3\}$. Suppose without loss of generality that we have $m_1 \le m_2 \le m_3$. Since $\phi(x)$ has finite support on the interval $0 < x < 1$, it is straightforward to show that

$$C(m_1,n_1,m_2,n_2,m_3,n_3) = 0 \quad \begin{array}{c} unless \\ and \\ (implies) \end{array} \quad \begin{array}{l} 2^{(m_2-m_1)}n_1 \le n_2 \le 2^{(m_2-m_1)}(n_1+1)-1 \\ 2^{(m_3-m_2)}n_2 \le n_3 \le 2^{(m_3-m_2)}(n_2+1)-1 \\ 2^{(m_3-m_1)}n_1 \le n_3 \le 2^{(m_3-m_1)}(n_1+1)-1 \end{array} \quad (14)$$

for any wavelet basis function $\phi(x)$. Usually (14) simplifies considerably (see Section C below).

Now insert the wavelet expansions in (12) into the layer stripping equations (4) (now regarding both x and t as continuous independent variables), and apply the wavelet transform (8a) twice, first in x, and then with x replaced by t, to the result. Using (12) and (13), this yields

$$D(m_1,n_1+2^{m_1}\Delta,m_2,n_2+2^{m_2}\Delta) = D(m_1,n_1,m_2,n_2)$$

$$-\sum_{m_3}\sum_{n_3}\sum_{m_5}\sum_{n_5} C(m_1,n_1,m_3,n_3,m_5,n_5)\Delta R(m_3,n_3)U(m_5,n_5,m_2,n_2); \quad (15a)$$

$$U(m_1,n_1+2^{m_1}\Delta,m_2,n_2-2^{m_2}\Delta) = U(m_1,n_1,m_2,n_2)$$

$$-\sum_{m_3}\sum_{n_3}\sum_{m_5}\sum_{n_5} C(m_1,n_1,m_3,n_3,m_5,n_5)\Delta R(m_3,n_3)D(m_5,n_5,m_2,n_2). \qquad (15b)$$

For small Δ we have $d(x,x) = 1$ in (4b) to second order in Δ ((11) also states this). Now set $t = x$ in the expansion (12b), apply the wavelet transform (8a), and use (12c) and (13). This yields

$$R(m_1,n_1)\Delta = \sum_{m_2}\sum_{n_2}\sum_{m_3}\sum_{n_3} C(m_1,n_1,m_2,n_2,m_3,n_3)U(m_3,n_3,m_2,n_2). \qquad (16)$$

Alternatively, we can avoid the (good) approximation $d(x,x) \approx 1$ by using an argument analogous to the one used to derive (9b). Recall that the wavelet basis function $\phi(x)$ has compact support on the interval $0 < x < 1$, and that by causality $d(x,t) = u(x,t) = 0$ for $t < x$. These imply that $D(m_1,n_1,m_2,n_2) = U(m_1,n_1,m_2,n_2) = 0$ for $2^{-m_2}n_2 < 2^{-m_1}(n_1 - 1)$ (this can be seen by considering the two-dimensional (x,t) plane; again the -1 comes from the length of the support of $\phi(x)$). Setting $n_2 = 2^{(m_2-m_1)}n_1$ in (15b) then yields a linear system of equations for $R(m_1,n_1)$ (note from (14) that $C(m_1,n_1,m_2,n_2,m_3,n_3)$ is nonzero only for certain n_i). However, this approach does not seem to be warranted, due to its complexity; we mention it only due to similarity to the computation of the matrix reflection coefficient in the multichannel Schur algorithm [15].

B. Comments

We comment on some distinctions between equations (15) and (9):

1. The most interesting aspect of (15) is that in the TWO-dimensional wavelet transform domain, the coupled wave system (4) (continuous in both the independent variables x and t) is AGAIN broken up into a set of coupled discrete-space discrete-time wave systems. The wave speed of the discrete wave system at resolutions m_1 (for x) and m_2 (for t) is $2^{(m_1-m_2)}$; this explains the causality relation noted above. This is quite remarkable, considering the nonlinearity of the problem in the variable x;
2. In terms of fast algorithms, (15) describes a coupled multichannel lattice structure, with coupling between different scales specified by the summation over m_5, and the coupling coefficient computed from $R(\cdot)$ and $C(\cdot)$ by the summations over m_3 and n_3.

The summation over n_3 has limited range (see (14)), and corresponds to mixing of the scales in the interval $(x, x+\Delta)$;

3. Initialization of (15) requires knowledge of $D(m_1, n_1, m_2, n_2)$ and $U(m_1, n_1, m_2, n_2)$ for $0 \le n_1 \le (2^{m_1} - 1)\Delta$. This corresponds to knowledge of $d(x,t)$ and $u(x,t)$ for $0 \le x \le \Delta$. Recall that the discretized Schur algorithm (4) physically reconstructs a discrete layered medium consisting of homogeneous layers of thicknesses Δ; thus we have $d(x,t) = k(t-x)$ and $u(x,t) = k(t+x)$ for $0 \le x \le \Delta$. Equations (15) can similarly be initialized if we assume a homogeneous medium *only for* $0 \le x \le \Delta$
4. In propagating (15) and (16), note that fractional values of the translation index n are required for scales m such that $2^m\Delta$ is not an integer (again it is suggested that $\Delta = 2^{-M}$ for some integer M). This has the same interpretation as in the time-only wavelet transform algorithm (see Section III), although now this applies for both space and time;
5. However, fractional values of n create a problem in propagating the *coupled* equations (15), since the nonzero values of the coupling function $C(\cdot)$ may require not-yet-computed function updates. To see this, consider two updates

$$D(m_1, n_1, m_2, n_2) \rightarrow D(m_1, n_1 + 2^{(m_1 - M)}, m_2, n_2 + 2^{(m_1 - M)})$$

$$D(m_3, n_3, m_4, n_4) \rightarrow D(m_3, n_3 + 2^{(m_3 - M)}, m_4, n_4 + 2^{(m_3 - M)}),$$

where $m_1 < M$. From (14), it is possible for $D(m_3, n_3, m_4, n_4)$ to be coupled to $D(m_1, n_3 + 1, m_2, n_2)$, which has not been computed yet since $2^{(m_1 - M)} < 1$.

C. Example: Haar Basis Function

To illustrate better the above points, and the operation of (15) and (16), we now specialize to the Haar wavelet basis function shown in Figure 3. This illustrates typical simplifications that occur when (14) and (15) are specialized to a specific basis function, and results in simpler equations.

First, we need to compute the function $C(m_1, n_1, m_2, n_2, m_3, n_3)$. Again, without loss of generality we assume $m_1 \le m_2 \le m_3$. It is straightforward to show that for the Haar

wavelet basis function we have

$$C(m_1,n_1,m_2,n_2,m_3,n_3)$$

$$\begin{aligned}
&= 2^{m_1/2} \quad if \quad m_1 < m_2 = m_3 \\
&\quad\quad and \quad 2^{(m_2-m_1)}n_1 \le n_2 = n_3 \le 2^{(m_2-m_1)}(n_1+1/2)-1; \\
&= -2^{m_1/2} \quad if \quad m_1 < m_2 = m_3 \\
&\quad\quad and \quad 2^{(m_2-m_1)}(n_1+1/2) \le n_2 = n_3 \le 2^{(m_2-m_1)}(n_1+1)-1; \\
&= 0 \quad\quad otherwise.
\end{aligned} \tag{17}$$

Note in particular that if $m_1 \neq m_2 \neq m_3$ then $C(\cdot) = 0$, and if $m_1 = m_2 = m_3$ then $C(\cdot) = 0$. Equations (15) then simplify to

$$D(m_1,n_1+2^{m_1}\Delta,m_2,n_2+2^{m_2}\Delta) = D(m_1,n_1,m_2,n_2)$$

$$-\sum_{\substack{m_3=0 \\ m_3\neq m_1}}^{\infty}\sum_{n_3=2^{(m_3-m_1)}n_1}^{2^{(m_3-m_1)}(n_1+1)-1} 2^{\pm MIN[m_1,m_3]/2}\Delta R(MAX[m_1,m_3],n)U(m_3,n_3,m_2,n_2)$$

$$-\Big(\sum_{m_3=0}^{m_1-1} 2^{\pm m_3/2}\Delta R(m_3,n_3=[2^{(m_3-m_1)}n_1])\Big)U(m_1,n_1,m_2,n_2); \tag{18a}$$

$$U(m_1,n_1+2^{m_1}\Delta,m_2,n_2-2^{m_2}\Delta) = U(m_1,n_1,m_2,n_2)$$

$$-\sum_{\substack{m_3=0 \\ m_3\neq m_1}}^{\infty}\sum_{n_3=2^{(m_3-m_1)}n_1}^{2^{(m_3-m_1)}(n_1+1)-1} 2^{\pm MIN[m_1,m_3]/2}\Delta R(MAX[m_1,m_3],n)D(m_3,n_3,m_2,n_2)$$

$$-\Big(\sum_{m_3=0}^{m_1-1} 2^{\pm m_3/2}\Delta R(m_3,n_3=[2^{(m_3-m_1)}n_1])\Big)D(m_1,n_1,m_2,n_2), \tag{18b}$$

where:

1. $n = n_1$ if $m_1 > m_3$ and $n = n_3$ if $m_1 < m_3$ in $R(MAX[m_1,m_3],n)$;
2. The sign in $2^{\pm MIN[m_1,m_3]/2}$ is chosen according to the value of n_3 (see (17));
3. For $m_3 < m_1$ the sum over n_3 becomes a single value $n_3 = [2^{(m_3-m_1)}n_1]$, where $[\cdot]$ denotes the greatest integer function.

Note the simplicity of the coupling between scales in (18), and how the coupling between a scale and itself is more complex. Also note that the coupling from a coarser scale (smaller m_1) to a finer scale (larger m_1) varies exponentially with the *smaller* scale. This suggests that one possible way of implementing (18) is to simply neglect the coupling

from scales with $m < M$ (where $\Delta = 2^{-M}$) to scales with $m > M$, since the coarser a scale is, the smaller its coupling to finer scales is (note that if $m < M$, then $2^m\Delta = 2^{(m-M)} < 1$). After propagating the finer scales with $m > M$, and having computing the values needed to propagate the coarser scales with $m < M$, these coarser scales can then be propagated.

D. The Born Approximation

The Born approximation is a linearization of the inverse scattering problem; the idea is to make the reflectivity function $r(x)$ be linearly related to the reflection response $k(x)$. One way to do this is to scale $r(x)$ by a small parameter ϵ, expand $d(x,t)$ and $u(x,t)$ in Taylor series in ϵ, and discard all terms of order ϵ^2 or smaller. This eliminates the nonlinear terms $r(x)d(x,t)$ and $r(x)u(x,t)$ in (1). The result is

$$\left(\frac{\partial}{\partial x} + \frac{\partial}{\partial t}\right) d(x,t) = 0 \rightarrow d(x,t) = d(x-t) \tag{19a}$$

$$\left(\frac{\partial}{\partial x} - \frac{\partial}{\partial t}\right) u(x,t) = 0 \rightarrow u(x,t) = u(x+t) \tag{19b}$$

and these in conjunction with (11) immediately yield

$$r(x) = 2u(x,x) = 2u(0,2x) = 2k(2x). \tag{20}$$

Equation (20) has a nice physical interpretation: the reflection response is due entirely to direct reflections of the probing impulse by the reflectivity function of the medium. Note that multiple scattering, which is inherently nonlinear, is neglected; for this reason the Born approximation is often called a single-scattering approximation.

Taking the wavelet transform (8) of (20) gives the very simple result

$$R(m,n) = \sqrt{2}K(m-1,n). \tag{21}$$

In the Born approximation, the wavelet transform of the reflectivity function equals the wavelet transform of the reflection response, on a scale one octave coarser. This is due to two-way traveltime–the reflectivity function $r(x)$ is imaged at time $t = 2x$ since the impulse must travel down to depth x and then back to the surface to be measured as $k(x)$.

V. WAVELET TRANSFORM IN SPACE AND TIME–INTEGRAL EQUATION

A. Derivation of Discrete Linear System of Equations

To obtain an alternative to the two-dimensional wavelet transform domain layer stripping algorithm of Section IV, we now apply the two-dimensional wavelet transform to the Krein integral equation (5). The result is a linear system of equations with block-slanted-Toeplitz structure (defined below), which relates $r(x)$ at different scales to $k(x)$ at different scales. Unlike the algorithm of Section 4, no approximations are necessary to use this result.

First, we make a minor change in the Krein integral equation (5). Define

$$k'(x) = 2k(2x); \quad h'(x,t) = 2h(x,2t-x); \quad t' = (x+t)/2; \quad z' = (x+z)/2. \tag{22}$$

Then (5) (multiplied by 2) and (6) can be rewritten as

$$k'(x-t') = h'(x,t') + \int_0^x h'(x,z')k'(|z'-t'|)dz'; \quad 0 \le t' \le x; \quad 0 \le x \le 1 \tag{23}$$

$$r(x) = h'(x,0), \tag{24}$$

respectively. This merely changes the interval $-x \le t \le x$ to the interval $0 \le t' \le x$, for both the range of integration and the range of validity of the integral equation. We further alter the former to $-\infty < t < \infty$ by simply defining $h'(x,t) = 0$ for $t < 0$ or $t > x$; this will prove useful below.

Now let the wavelet basis function $\phi(x)$ have the properties listed at the beginning of Section 4. Define the following two-dimensional wavelet expansions:

$$H'(m_1,n_1,m_2,n_2) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h'(x,t)2^{m_1/2}\phi(2^{m_1}x-n_1)2^{m_2/2}\phi(2^{m_2}t-n_2)\,dx\,dt;$$

$$h'(x,t) = \sum_{m_1}\sum_{n_1}\sum_{m_2}\sum_{n_2} H'(m_1,n_1,m_2,n_2)2^{m_1/2}\phi(2^{m_1}x-n_1)2^{m_2/2}\phi(2^{m_2}t-n_2); \tag{25a}$$

$$K'(m_3,n_3) = \int_{-\infty}^{\infty} k'(x)2^{m_3/2}\phi(2^{m_3}x-n_3)\,dx; \quad k'(x) = \sum_{m_3}\sum_{n_3} K'(m_3,n_3)2^{m_3/2}\phi(2^{m_3}x-n_3). \tag{25b}$$

Also define the discrete function (note the difference from (13))

$$E(m_1,n_1,m_2,n_2,m_3,n_3) =$$

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} 2^{m_1/2}\phi(2^{m_1}x-n_1)2^{m_2/2}\phi(2^{m_2}y-n_2)2^{m_3/2}\phi(2^{m_3}(x-y)-n_3)\,dx\,dy. \quad (26)$$

Defining $x' = x - 2^{-m_1}n_1$ and $y' = y - 2^{-m_2}n_2$, we can rewrite (26) as

$$E(m_1,n_1,m_2,n_2,m_3,n_3) =$$

$$2^{(m_1+m_2+m_3)/2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\phi(2^{m_1}x')\phi(2^{m_2}y')\phi\Big(2^{m_3}(x'-y')-n_3+2^{m_3}(2^{-m_1}n_1-2^{-m_2}n_2)\Big)$$
$$= E(m_1,m_2,m_3,n_3,(2^{-m_1}n_1-2^{-m_2}n_2)). \quad (27)$$

Unlike $C(m_1,n_1,m_2,n_2,m_3,n_3)$, $E(m_1,m_2,m_3,n_3,(2^{-m_1}n_1-2^{-m_2}n_2))$ is not symmetric with respect to permutations of the subscripts $\{1,2,3\}$. However, it does have the *slanted-Toeplitz structure*, defined as being a function not of n_1 and n_2 separately, but a function of only their weighted difference $2^{-m_1}n_1 - 2^{-m_2}n_2$.

B. Linear System of Equations

Now insert the wavelet expansions in (25) into the modified Krein integral equation (23), and apply the wavelet transform (8a) twice, first in x, and then with x replaced by t, to the result. Using (25)-(27), this yields

$$K''(m_1,m_2,(2^{-m_1}n_1-2^{-m_2}n_2)) = H'(m_1,n_1,m_2,n_2)$$
$$+\sum_{m_3=0}^{M}\sum_{n_3} H'(m_1,n_1,m_3,n_3)K''(m_3,m_2,(2^{-m_3}n_3-2^{-m_2}n_2));$$
$$0\le n_1\le 2^{m_1}-1;\quad 0\le 2^{-m_2}n_2\le 2^{-m_1}n_1-1;\quad 0\le m_1,m_2\le M, \quad (28)$$

where $K''(\cdot)$ is defined from (25b) and (27) as

$$K''(m_1,m_2,(2^{-m_1}n_1-2^{-m_2}n_2))$$
$$=\sum_{m_3=0}^{M}\sum_{n_3=0}^{2^{m_3}-1} E(m_1,m_2,m_3,n_3,(2^{-m_1}n_1-2^{-m_2}n_2))K'(m_3,n_3), \quad (29)$$

and M is the (arbitrarily large) finest scale used in the wavelet expansion. The summation over n_3 in (28) is over all integer values of n_3 such that

$$0 \leq 2^{-m_3} n_3 \leq 2^{-m_1} n_1 - 1. \tag{30}$$

The system matrix specified by $K''(\cdot)$ in the system of equations (28) has a block-slanted-Toeplitz structure, in that in the $(m_1, m_2)^{th}$ block the elements (n_1, n_2) are equal for constant values of $2^{-m_1} n_1 - 2^{-m_2} n_2$, i.e., along diagonals of slope $2^{(m_2 - m_1)}$. Note that matrix coordinates start at $(0,0)$ here, not $(1,1)$. In fact, the system matrix has the following structure (letters denote equal entries, excluding symmetry):

$$\begin{bmatrix} * & \begin{bmatrix} * & * \end{bmatrix} & \begin{bmatrix} * & * & * & * \end{bmatrix} & \cdots \\ \begin{bmatrix} * \\ * \end{bmatrix} & \begin{bmatrix} a & * \\ * & a \end{bmatrix} & \begin{bmatrix} b & f & * & * \\ * & * & b & f \end{bmatrix} & \cdots \\ \begin{bmatrix} * \\ * \\ * \\ * \end{bmatrix} & \begin{bmatrix} b & * \\ f & * \\ * & b \\ * & f \end{bmatrix} & \begin{bmatrix} c & d & e & * \\ d & c & d & e \\ e & d & c & d \\ * & e & d & c \end{bmatrix} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{31}$$

C. Comments

1. To reconstruct the wavelet transform $R(m,n)$ of $r(x)$ from the wavelet transform $K(m,n)$ of $k(x)$, simply do the following:
   a. Compute $K''(\cdot)$ from $k(x)$ using (22), (25b), and (29);
   b. Solve the linear system of equations (28) for $H'(\cdot)$;
   c. Compute $R(m,n)$ from $H'(\cdot)$ using

$$R(m_1, n_1) = \sum_{m_2=0}^{M} H'(m_1, n_1, m_2, n_2 = 0)\phi(0^+) \tag{32}$$

   which is derived by applying the wavelet transform (8a) to (24) while setting $t = 0$ in the wavelet expansion (25a). The system of equations (28), along with (29) and (32), thus relates the wavelet transforms $K(m,n)$ and $R(m,n)$;
2. The limits on the indices in (28) follow directly from the limits $0 \leq x \leq 1$, $0 \leq t \leq x$ in (23), and from the finite support of $\phi(x)$ on $0 < x < 1$. Equation (30) follows from the same reasoning used in Section IV;

3. Unlike the algorithms of Sections III and IV, there are no fractional values of the n_i, since there is no Δ parameter;
4. $k(|z-t)|$ is under the integral in (23), and $k(x-t)$ on the left side is only used for $x-t>0$. Since both $k(x)$ and $\phi(x)$ are causal functions, we have $k(|x|)=k(x)+k(-x)$, and similarly for $\phi(x)$. As a result, $(x-y)$ in (26) and (27) should be replaced by $|x-y|$, so that $E(\cdot)$ is symmetric with respect to permutations of the indices $\{1,2\}$. This means the system of equations (28) is symmetric, as is the integral equation (23);
5. If the Krein integral equation (23) is discretized with $\Delta = 2^{-M}$, then the resulting system of equations has size 2^M. If the finest scale used in (28) is M, then (31) shows that the size of the system (28) is

$$1+2+4+8+\ldots+2^M = 2^{M+1}-1 \tag{33}$$

so (28) is only twice as large as a simple discretization of (23);
6. The kernel $K''(\cdot)$ (28) has the displacement structure

$$K''(m_1,n_1+2^{m_1}\Delta,m_2,n_2+2^{m_2}\Delta)-K''(m_1,n_1,m_2,n_2)=0; \tag{34}$$

this is clearly associated with the coupled wave systems (15), just as the Toeplitz structure in the discretized Krein integral equation is associated with discrete transmission lines [12]. The only difference is the differing wave speeds in different wave systems. However;
7. The form of (31) makes it clear why the multichannel coupled lattice structure (15) derived in Section IV requires coupling to not-yet-computed quantities at different scales. The block-slanted-Toeplitz property imposes more constraints on blocks near the main block-diagonal, while blocks far away from the main block-diagonal have less structure. In short, although there is some structure in (31) (enough to associate lattice equations (15) with it), there is ***not enough*** structure to allow the existence of a fast algorithm!

VI. CONCLUSION

The wavelet transform has been used to generate a multiresolution analysis of the exact (all multiple reflections included) one-dimensional inverse scattering problem. Three

new wavelet transform domain algorithms have been derived for this problem. The first operates in the one-dimensional (time-only) wavelet transform domain. It consists of a decoupled set of discrete wave equations, each with a different wave speed determined by the wavelet transform resolution. Any one scale is sufficient to reconstruct the reflectivity function; this permits some flexibility (e.g., using several scales for noise reduction) in implementing this algorithm.

The second and third algorithms operate in the two-dimensional (time and space) wavelet transform domain. One consists of a coupled multichannel set of discrete wave equations; however, an approximation of neglecting some coupling between scales is required to use it. An example using the Haar wavelet basis function suggests this is not unreasonable. The other algorithm consists of solving a block-slanted-Toeplitz linear system of equations, which specifies directly the relation between the wavelet transforms of the reflectivity function and the impulse reflection response. The structure in this system of equations explains the form of the coupled multichannel set of discrete wave equations derived earlier. Finally, we note in passing that these results can be applied to the linear prediction problem of computing the linear least-squares estimation filter of a zero-mean wide-sense-stationary random process from its covariance function, since the Krein integral equation has the same form as a Wiener-Hopf integral equation.

ACKNOWLEDGMENTS

The author would like to thank Poovendran Raadhakrishnan for much useful discussion on this material. This work was supported by the Office of Naval Research under grant #N00014-90-J-1897.

REFERENCES

1. K.P. Bube and R. Burridge, "The one-dimensional inverse problem of reflection seismology," *SIAM Review*, vol. 25, pp. 497-559, 1983.
2. B. Gopinath and M.M. Sondhi, "Determination of the shape of the human vocal tract from acoustical measurements," *Bell System Tech. J.*, vol. 49, pp. 1195-1214, 1970.

3. B. Gopinath and M.M. Sondhi, "Inversion of the telegraph equation and the synthesis of nonuniform lines," *Proc. of IEEE*, vol. 59, pp. 383-392, 1971.
4. M.G. Krein, "On a method for the effective solution of the inverse boundary value problem," *Dokl. Akad. Nauk USSR*, vol. 94, pp. 987-990, 1954.
5. A.E. Yagle and B.C. Levy, "Application of the Schur algorithm to the inverse problem for a layered acoustic medium," *J. Acoust. Soc. Am.*, vol. 76, pp. 301-308, 1984.
6. A.M. Bruckstein, T. Kailath, and B.C. Levy, "Differential methods in inverse scattering," *SIAM J. Appl. Math.*, vol. 45, pp. 312-335, 1985.
7. O. Rioul and M. Vetterli, "Wavelets and signal processing,"*IEEE Sig. Proc. Mag.*, vol. 8, pp. 14-38, Oct. 1991.
8. Special Issue, *IEEE Trans. Info. Th.*, vol. IT-38, March 1992.
9. I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Comm. in Pure and Appl. Math.*, vol. 41, pp. 909-996, 1988.
10. S. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 11, pp. 674-693, 1989.
11. T. Kailath, "A theorem of I. Schur and its impact on modern signal processing," *Operator Theory, Advances and Applications*, vol. 18, 1986.
12. A.M. Bruckstein and T. Kailath, "Inverse scattering for discrete transmission-line models," *SIAM Review*, vol. 29, pp. 359-389, 1987.
13. T. Kailath, L. Ljung, and M. Morf, "Generalized Krein-Levinson equations for efficient calculation of Fredholm resolvents of nondisplacement kernels," in *Topics in Functional Analysis, Advances in Mathematics Supplementary Studies 3*, Academic Press, New York, 1978, pp. 169-184.
14. I. Daubechies, "The wavelet transform, time-frequency localization and signal analysis," *IEEE Trans. Info. Th.*, vol. IT-36, pp. 961-1005, 1990.
15. J. Rissanen, "Algorithms for triangular decomposition of block Hankel and Toeplitz matrices with application to factoring positive matrix polynomials, *Math. Comput.*, vol. 27, pp. 147-154, 1973.

FIGURE HEADINGS

1. A differential section of the scattering medium to be reconstructed.
2. The sampling grid (scale vs. translation) for the wavelet transform.
3. The Haar wavelet basis function.

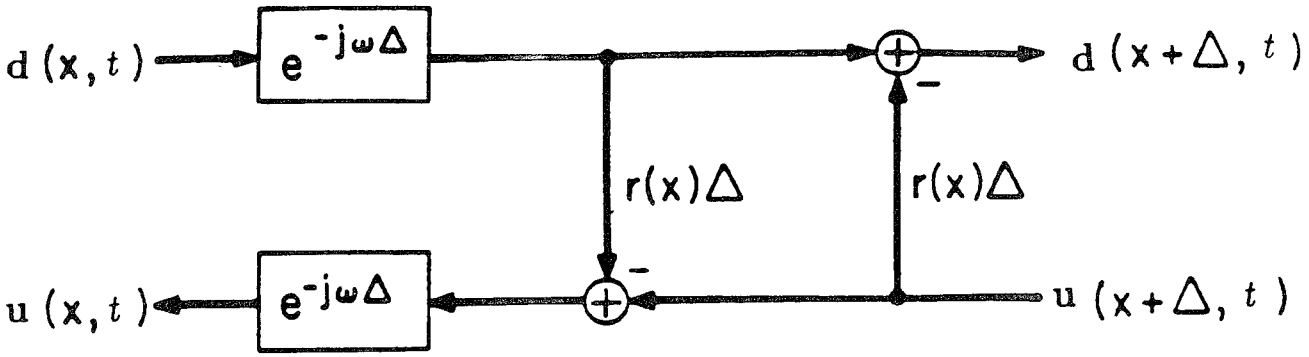


FIG. 1

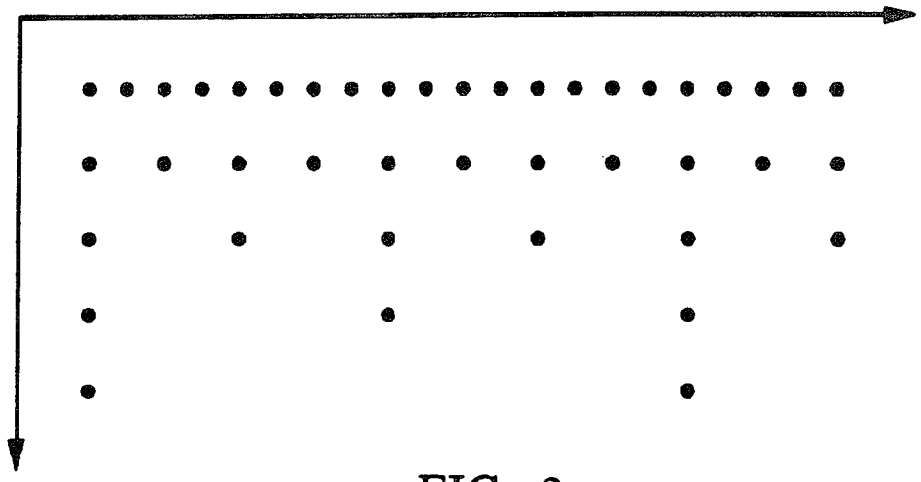


FIG. 2

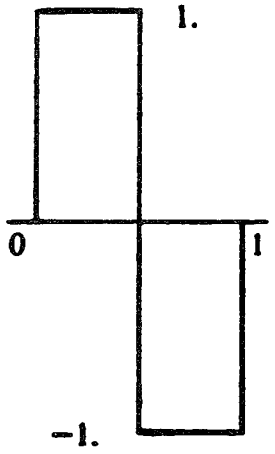


FIG. 3

APPENDIX D

A.E. Yagle, "Inversion of Spherical Means Using Geometric Inversion and Radon Transforms," Inverse Problems 8(6), 949-964, December 1992.

This paper analyzes the problem of reconstructing a function from its spherical means passing through the origin. A new application of this problem to diffraction tomography is noted: We show that given probing by impulsive plane waves at all angles of incidence, only a single receiving sensor is necessary, not an array of sensors.

Two versions of the problem are defined. A layer-stripping-type algorithm is derived for one version (we use the term invariant imbedding in the paper, since it is more familiar to readers, but it is really layer stripping). The two versions are shown to be equivalent to the usual and exterior inverse Radon transforms, respectively, using geometric inversion (reflection about a circle). A simple numerical example is also included.

8 (1992) 949–964. Printed in the UK

Inversion of spherical means using geometric inversion and Radon transforms

A E Yagle

Department of Electrical Engineering and Computer Science, The University of Michigan, Ann Arbor, MI 48109-2122, USA

Received 8 May 1991, in final form 23 June 1992

Abstract. We consider the problem of reconstructing a continuous function on $\mathcal{R}^n$ from certain values of its spherical means. A novel aspect of our approach is the use of geometric inversion to recast the inverse spherical mean problem as an inverse Radon transform problem. We define two spherical mean inverse problems: the entire problem and the causal problem. We then present a dual filtered backprojection algorithm for the entire problem and an invariant imbedding algorithm for the causal problem. We then show how geometric inversion can be used to transform the entire and causal problems into complete and exterior inverse Radon transform problems, respectively. We also consider the uniqueness problem, for which we prove a sufficiency theorem and we note an application of these results to diffraction tomography.

1. Introduction

1.1. The basic problem

We consider the following problem. Given values of the 'spherical means'

$$\hat{f}(t,e)=\int_{|x|=te\cdot e_x} f(x)\,d\sigma=\int_S f(|x|,e_x)\frac{|x|^{n-1}}{e\cdot e_x}\bigg|_{|x|=te\cdot e_x} de_x \quad (1.1)$$

of a function $f(x)=f(|x|,e_x)$ defined on $x\in\mathcal{R}^n$, determine $f(x)$ from $\hat{f}(t,e)$. Here e and e_x are unit vectors in $\mathcal{R}^n$, $d\sigma=|x|^{n-1}/(e\cdot e_x)de_x$ is differential surface area on the surface of the sphere $|x|=te\cdot e_x$ in $\mathcal{R}^n$, and S is the unit half-sphere $e\cdot e_x\geq 0$ in $\mathcal{R}^n$ (although the orientation of S actually depends on e, this will not be explicitly indicated). Note that $|x|=te\cdot e_x$ describes a sphere in $\mathcal{R}^n$ that passes through the origin, has diameter t and centre at the point $te/2$ in $\mathcal{R}^n$. Also note that if $f(x)$ is a continuous function, $\hat{f}(t,e)$ will also be a continuous function in both t and e.

This problem of inverting spherical means has been treated by many authors, e.g. [1, 2]. However, the usual problem is to recover $f(x)$ from its integrals over spheres whose centres all lie on a plane; only the even part of $f(x)$ with respect to this plane can be recovered. Nonetheless, this problem is useful in reconstructing a scattering acoustic medium from its responses to an impulsive point source located at various places on its surface; in the Born approximation these responses are simply the spherical means of the medium reflectivity [3, 4].

0266-5611/92/060949+16$07.50 © 1992 IOP Publishing Ltd

The problem we investigate is to recover $f(x)$ from its integrals over spheres passing through the origin. This problem is of interest in diffraction tomography; we present a novel application of the results of this paper to diffraction tomography in subsection 1.3 below. The novelty of our approach is that we use geometric inversion to transform the inverse problem (1.1) into an inverse Radon transform problem. By reflecting functions across a given sphere, integrals over spheres are transformed into integrals over planes.

Specifically, we define two spherical mean inverse problems: the entire problem of reconstructing $f(x)$ from $\{\hat{f}(t,e)\}$ for all t and e; and the causal problem of reconstructing $\{f(x),|x|<t\}$ from $\{\hat{f}(\tau,e),|\tau|<t\}$ for any $t>0$. We then present a dual filtered backprojection algorithm for the entire problem, and an invariant imbedding algorithm for the causal problem. We then show how geometric inversion can be used to transform the entire and causal problems into complete and exterior inverse Radon transform problems, respectively. Finally, we state and prove a sufficiency theorem on the inversion of (1.1).

1.2. Review of diffraction tomography

In this section we quickly review the basic result of diffraction tomography. This allows us to develop in subsection 1.3 a new application of the spherical mean inverse problem. The material in subsection 1.2 is taken from [5].

The basic two-dimensional (2D) diffraction tomography problem is formulated as follows. A 2D acoustic medium has constant wave speed c_0, except in a region of finite extent in which the wave speed is a 2D function $c(r)\neq c_0$ of position $r=(x,y)\in\mathcal{R}^2$. An infinite plane wave $u_0(t-s_0\cdot r/c_0)$, propagating in the direction specified by the unit vector $s_0=(\cos\phi_0,\sin\phi_0)$, is used to probe the inhomogeneous region. The scattered field $d(r,t;s_0)$ is measured for all time t by an array of sensors along the line $x=l_0$. The goal is to reconstruct $c(r)$ from the measurements $d(r=(l_0,y),t;s_0)$ of the scattered field, for all positions y along the line of sensors, time t, and directions of incidence s_0. Note that the problem is overdetermined we are using data with three degrees of freedom to reconstruct a function with two degrees of freedom. The problem is illustrated in figure 1.

This problem is of interest in offset vertical seismic profiling and ultrasound medical tomography. In both cases, the wavelengths of the incident wave field $u_0(t)$ and scattered wave field $d(r,t;s_0)$ are long enough such that diffraction effects cannot be neglected.

First, define the scattering potential

$$o(r)=1-(c_0/c(r))^2. \quad (1.2)$$

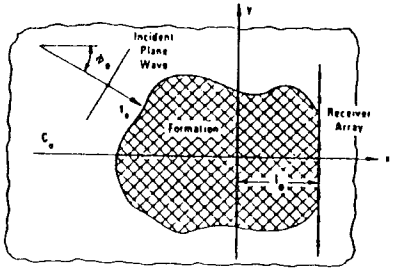


Figure 1. The basic diffraction tomography problem with a fixed array.

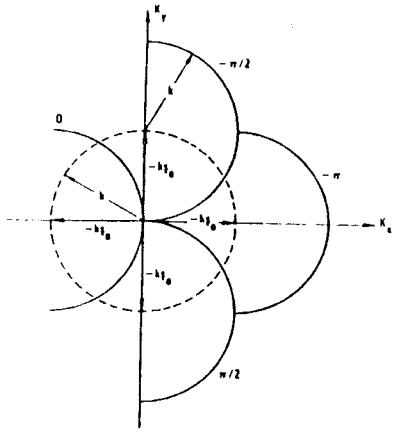


Figure 2. Some of the semi-circles on which the spatial Fourier transform $O(K)$ of scattering potential is reconstructed in the diffraction tomography problem.

Note that $o(r) = 0$ outside the inhomogeneous region. Then the temporal Fourier transform $U(r, k; s_0)$ of the total wave field satisfies

$$(\nabla^2 + k^2)U(r, k; s_0) = k^2 o(r)U(r, k; s_0) \tag{1.3}$$

where $k = \omega/c_0$ is the wavenumber.

Under the assumption that $o(r)$ is small (weak scattering), we may use either the Born or Rytov approximations. Define the temporal Fourier transforms of the incident and scattered wave fields as $U_0(k)$ and $D(r, k; s_0)$, where

$$D(r, k; s_0) = \frac{2i}{k^2} \begin{cases} U(r, k; s_0)/U_0(k) - e^{ik s_0 \cdot r} & \text{Born} \\ (\log[U(r, k; s_0)/U_0(k)] - ik s_0 \cdot r)e^{ik s_0 \cdot r} & \text{Rytov.} \end{cases} \tag{1.4}$$

Also define the 2D spatial Fourier transform of the scattering potential as $O(K) = \mathcal{FF}_{r \to K}\{o(r)\}$, and the 1D spatial Fourier transform of the measurements $D(r, k; s_0)$ along the array of sensors $r = (l_0, y)$ as

$$\tilde{D}(k', k; s_0) = \int_{-\infty}^{\infty} D(r = (l_0, y), k; s_0)e^{ik'y}\, dy. \tag{1.5}$$

Then it is straightforward to show (see [5]) that the known $\tilde{D}(k', k; s_0)$ and the unknown $O(K)$ are related by

$$\tilde{D}(k', k; s_0) = \frac{e^{i\sqrt{k^2 - k'^2}\, l_0}}{\sqrt{k^2 - k'^2}} O(K = k(s - s_0)) \tag{1.6}$$

where the unit vector s is defined as

$$s = (\cos\theta, \sin\theta) \qquad \sin\theta = k'/k \qquad \cos\theta = \sqrt{1 - (k'/k)^2} \qquad |\theta| < \pi/2. \tag{1.7}$$

Equation (1.6) shows that the measurements $d(r = (l_0, y), t; s_0)$ of the scattered field specify the Fourier transform $O(K)$ of the scattering potential on semi-circles of radius k centred on $-ks_0 = (-k\cos\phi_0, -k\sin\phi_0)$, where ϕ_0 is the angle of incidence of the probing plane wave (see figure 1). This is illustrated in figure 2 for $\phi_0 = 0, \pi/2, \pi$ and $3\pi/2$.

Three comments are in order here. First, why do we get semi-circles instead of full circles? This occurs because evanescent waves (waves which decay exponentially with distance instead of propagating) cannot be measured in practice. This restricts $|k'| < k$ (see [5]), which from (1.7) restricts $|\theta| < \pi/2$. Second, in the high-frequency limit $k \to \infty$ the semi-circles become lines, so the diffraction tomography problem becomes an inverse Radon transform problem, and (1.6) becomes the projection-slice theorem. Third, it is clear that without loss of generality we may set $l_0 = 0$; the only occurrence of l_0 is in the factor $e^{i\sqrt{k^2 - k'^2}\, l_0}$ in (1.6), which we recognize as the migration operator between the lines $x = l_0$ and $x = 0$.

1.3. A new application of the spherical mean inverse problem in diffraction tomography

In this section we show, for the first time, that measurements along an array of sensors are not necessary. In fact, measurements at a single sensor are sufficient to reconstruct the scattering potential in either the Born or Rytov approximations provided either: (1) truly wideband data are available; or (2) the average value of the potential is known. We show this by formulating this problem as the spherical mean inverse problem considered in the rest of the current paper.

The problem we consider in this section is identical to the problem considered in subsection 1.2, except that now the scattered field $d(r, t; s_0)$ is measured only at the origin $r = (0, 0)$, instead of along the line $\{r = (l_0, y), -\infty < y < \infty\}$. Clearly there is no loss of generality in taking the origin to be the site of measurements. From (1.5), this amounts to measuring

$$D((0,0), k; s_0) = \frac{1}{2\pi} \int_{-k}^{k} \tilde{D}(k', k; s_0)\, dk' \tag{1.8}$$

since $\tilde{D}(k', k; s_0)$ for $|k'| > k$ are the evanescent components of the scattered field which are negligible. Inserting (1.6) in (1.8), setting $l_0 = 0$, and changing variables from k' to θ using (1.7) (the denominator $\sqrt{k^2 - k'^2}$ in (1.6) disappears in the variable change) shows that the measurements are actually

$$\begin{aligned} D((0,0), k; s_0) &= \frac{1}{2\pi} \int_{-\pi/2}^{\pi/2} O(K = k(\cos\theta - \cos\phi_0, \sin\theta - \sin\phi_0))\, d\theta \\ &= \frac{1}{2\pi} \int_{-\pi/2}^{\pi/2} O\left(|K| = 2k\sin\left(\frac{\theta - \phi_0}{2}\right)\right) d\theta \\ &= \frac{1}{2\pi} B\{O(K)\}(t, \theta) \qquad |\theta| < \pi/2. \end{aligned} \tag{1.9}$$

The backprojection operator $B\{\cdot\}$ is defined in (2.2) below; it is essentially the spherical mean operator. The last equality in (1.9) comes from noting that the semi-circle $K = k(s - s_0)$ has diameter $2k$ and is centred at $-ks_0 = (-k\cos\phi_0, -k\sin\phi_0)$. Changing ϕ_0 to $\phi_0 + \pi$ transforms the sine in the third term of (1.9) into a cosine.

Equation (1.9) states that $D((0,0), k; s_0)$ are essentially the semi-circular means of $O(K)$; this is evident from (1.8), which integrates along each of the semi-circles in figure 2. To transform the semi-circular means into full-circular means we need the 'other halves' of the semi-circles in figure 2.

To obtain these 'other halves,' we simply note that since $o(r)$ is real, $O(-K) = O^*(K)$, where $*$ denotes complex conjugate. Since the integral of complex conjugates is the complex conjugate of the integral, the semi-circular means of $O(-K)$ will be the complex conjugates of the semi-circular means of $O(K)$. But the right-semi-circular means of $O(-K)$ are the same as the left-semi-circular means of $O(K)$. Therefore, a complete set of full-circular means of $O(K)$ can be obtained by simply adding $D((0,0),k;s_0)$ and $D^*((0,0),k;-s_0)$, for each k and s_0.

We can then reconstruct $O(K)$, and then scattering potential $o(r)$ and wave speed $c(r)$, from these circular means using the methods proposed in the rest of this paper. If truly wideband ($0 < k < \infty$) data are available, we may use the procedure of subsection 2.3 below. Otherwise, we must know $O(K)$ for small $|K|$; this low-wavenumber information corresponds roughly to the average value of $o(r)$.

The new result of this section is not surprising, from a degrees-of-freedom point of view. $o(r)$ has two degrees of freedom, while the measurements $d(r = (l_0, y), t; s_0)$ in subsection 1.2 have three degrees of freedom, so clearly there is some redundancy. This section has shown how this redundancy can be removed.

1.4. Geometric inversion

Let re be a point in $\mathcal{R}^n$ located a distance r from the origin in the direction specified by the unit vector e. The geometric inversion of re across the sphere of radius T in $\mathcal{R}^n$ is the point $(T/r)e$.

Consider the sphere $|x| = te \cdot e_x$ in $\mathcal{R}^n$, which passes through the origin and has diameter t and centre at the point $te/2$. The geometric inversion of this sphere across the sphere of radius T in $\mathcal{R}^n$ is $T/|x| = te \cdot e_x$, which can be rewritten as $e \cdot x = T/t$, which is a hyperplane passing within a distance T/t of the origin and having a normal unit vector e.

This can be seen more clearly in $\mathcal{R}^2$. Let (r, θ) and (x, y) be the polar and rectangular coordinates, respectively, of a point in $\mathcal{R}^2$. The geometric inversion of the circle $r = t\cos\theta$ (which passes through the origin and has diameter t) across the circle $r = T$ is $T/r = t\cos\theta$, which can be rewritten as $x = r\cos\theta = T/t$, which is a line perpendicular to the x-axis. Other circles passing through the origin can be similarly handled by first rotating coordinates.

2. Radon transforms, backprojection and causality

2.1. Causality properties of transforms

Recall that the 'Radon transform' of $f(x), x \in \mathcal{R}^n$ is defined as

$$\mathcal{R}\{f(x)\}(t,e) = \int_0^\infty \int_S f(|x|, e_x)\delta(t - |x|e \cdot e_x)|x|^{n-1}\,de_x\,d|x| \tag{2.1}$$

where e is a unit vector in $\mathcal{R}^n$. Note that (t, e) should not be regarded as a point in $\mathcal{R}^n$, since $\mathcal{R}\{f(x)\}(0, e_1) \neq \mathcal{R}\{f(x)\}(0, e_2)$ if $e_1 \neq e_2$; however, it is true that $\mathcal{R}\{f(x)\}(t, e) = \mathcal{R}\{f(x)\}(-t, -e)$.

Since the Radon transform of a function is its integral over the planes $t = e \cdot x$, the Radon transform has the following anti-causality property: For any $T > 0$, $\mathcal{R}\{f(x)\}(T, e)$ depends only on the values $\{f(x), |x| \geq T\}$. This is clear since the minimum distance between the origin and the plane $T = e \cdot x$ is T.

The 'backprojection transform' of $f(x), x \in \mathcal{R}^n$ is defined as

$$\mathcal{B}\{f(x)\}(t,e) = \int_S f(|x| = te \cdot e_x, e_x)\,de_x \tag{2.2}$$

where e and e_x are unit vectors in $\mathcal{R}^n$. $\mathcal{B}$ is half the adjoint operator to $\mathcal{R}$ ([7], p 115). The half is due to the integration over the half-sphere S; since $(|x|, e)$ and $(-|x|, -e)$ represent the same point in $\mathcal{R}^n$, $f(|x|, e) = f(-|x|, -e)$, and the integral over the opposing half-sphere equals the integral over S.

$$\int_{e \cdot e_x < 0} f(|x| = te \cdot e_x, e_x)\,de_x$$
$$= \int_{e \cdot e_x > 0} f(|x| = -te \cdot e_x, -e_x)\,de_x = \int_{e \cdot e_x > 0} f(|x| = te \cdot e_x, e_x)\,de_x.$$

More importantly, we have from (1.1) that

$$\hat{f}(t,e) = \int_S f(|x|, e_x)\frac{|x|^{n-1}}{e \cdot e_x}\bigg|_{|x| = te \cdot e_x} de_x = t\mathcal{B}\{|x|^{n-2}f(x)\} \tag{2.3}$$

so that inverting the spherical mean operator (1.1) is essentially equivalent to inverting the backprojection operator (2.2). Indeed, we have

$$f(x) = \frac{1}{|x|^{n-2}}\mathcal{B}^{-1}\{\tfrac{1}{t}\hat{f}(t,e)\} = \mathcal{B}^{-1}\{\tfrac{1}{t}\hat{f}(t,e)\} \qquad \text{if } n = 2. \tag{2.4}$$

Therefore, in the following, any results on inverting the backprojection operator will be directly applicable to inverting the spherical mean operator.

A simple example illustrating the difference between the two operators is in order here. Let $n = 3$ and $f(x) = 1$ everywhere in $\mathcal{R}^3$. Then $\mathcal{B}\{f(x)\} = \int_S de_x = 2\pi$ (the surface area of the half-sphere S) and $\hat{f}(t,e) = 4\pi(t/2)^2 = \pi t^2$ (the surface area of a sphere with radius $t/2$). For this example (2.3) becomes

$$\hat{f}(t,e) = t\mathcal{B}\{|x|\} = t\int_S te \cdot e_x\,de_x = t^2\int_0^{2\pi}\int_0^{\pi/2}\cos\theta\sin\theta\,d\theta\,d\phi = \pi t^2 \tag{2.5}$$

where θ is polar angle measured from direction e, $e \cdot e_x = \cos\theta$, and $de_x = \sin\theta\,d\theta\,d\phi$.

Inversion of both operators will be aided by the following causality property: For any $T > 0$, $\hat{f}(T, e)$ depends only on the values $\{f(x), |x| \leq T\}$. This is clear since the maximum distance between the origin and the sphere $|x| = Te \cdot e_x$ is T. Another way to see this is to note that backprojection at the point (T, e) can also be viewed as the average over all planes (in (t, e)) $|x| = te \cdot e_x$ passing through (T, e); any such plane must pass closer to the origin than T, so that for any such plane the minimum distance from the origin is $|x| < T$.

We show in section 3 that $\{f(x), |x| < T\}$ may be reconstructed from $\{\hat{f}(t,e), t < T\}$, provided that $f(x)$ is known in an arbitrarily small sphere centred at the origin; this is the causal spherical mean inverse problem. However, the availability of the additional data $\{\hat{f}(t,e), t > T\}$ will be shown to alter the poorly-conditioned causal spherical mean inverse problem into the relatively well-conditioned entire spherical mean inverse problem.

2.2. *The exterior Radon transform problem*

The anti-causality property of the Radon transform implies that it should be possible to reconstruct $\{f(x), |x| > t\}$ from its Radon transform values $\{\mathcal{R}\{f(x)\}(\tau, e), |\tau| > t\}$, for any $t > 0$. This is the exterior Radon transform problem. For $n = 2$ it may be solved as follows [6].

Assume $f(x)$ has finite support, so that there exists T such that $f(x) = 0$ for all $|x| > T$. Then $\mathcal{R}\{f(x)\}(\tau, e) = 0$ for $|\tau| > T$. Let t be any number such that $0 < t < T$. Both the unknown function $f(x) = f(|x|, \theta)$ and its known Radon transform $\mathcal{R}\{f(x)\}(\tau, \phi)$ (the projection angle ϕ replaces the unit vector $e = (\cos\phi, \sin\phi)$ for $n = 2$) are periodic in their angular variables θ and ϕ, respectively. Hence both may be expanded in Fourier series in their angular variables, yielding

$$f(x) = f(|x|, \theta) = \sum_{m=-\infty}^{\infty} F_m(|x|) e^{im\theta} \tag{2.6a}$$

$$\mathcal{R}\{f(x)\}(\tau, \phi) = \sum_{m=-\infty}^{\infty} RF_m(\tau) e^{im\phi} \tag{2.6b}$$

where

$$F_m(|x|) = \frac{1}{2\pi} \int_0^{2\pi} f(|x|, \theta) e^{-im\theta} \, d\theta \tag{2.7a}$$

$$RF_m(\tau) = \frac{1}{2\pi} \int_0^{2\pi} \mathcal{R}\{f(x)\}(\tau, \phi) e^{-im\phi} \, d\phi. \tag{2.7b}$$

Then the unknown function harmonics $\{F_m(|x|), t < |x| < T\}$ may be computed from the known projection harmonics $\{RF_m(\tau), t < \tau < T\}$ using the formula [6]

$$F_m(|x|) = -\frac{1}{\pi} \int_{|x|}^{T} \frac{\frac{d}{d\tau} RF_m(\tau) P_m(\tau/|x|)|x|}{\sqrt{\tau^2 - |x|^2}} \, d\tau. \tag{2.8}$$

where $P_m(\tau) = \cos(m \cos^{-1} \tau)$ is the Chebyschev polynomial of the first kind of order m.

It is clear from (2.8) that $\{f(x), |x| > t\}$ can be reconstructed in principle from $\{\mathcal{R}\{f(x)\}(\tau, \phi), |\tau| > t\}$. However, there are practical difficulties in using (2.8), since $P_m(\tau) \simeq (1/2)(2\tau)^m$ for large τ and m, so that evaluation of the integral (2.8) will be numerically unstable for $|x| \to 0$ or $m \to \infty$. Hence the exterior Radon transform problem is ill-posed, and it must be regularized. One way to do this follows.

First, we must assume that the function $f(x)$ is angularly bandlimited, so that its Fourier expansion (2.6a) has only a finite number of terms. This amounts to assuming $F_m(|x|) = 0$ for $|m| > M$, for some M. The Radon transform $\mathcal{R}\{f(x)\}$ will then also be angularly bandlimited, so that $RF_m(\tau) = 0$ for $|m| > M$. Then (2.8) need only be applied for the $2M + 1$ values of m such that $|m| \leq M$. Since only a finite number of projection angles will be available in practice, this assumption is not unreasonable.

Second, we must accept that we cannot completely reconstruct $f(x)$ from $\mathcal{R}\{f(x)\}$ using (2.8); rather, we can only reconstruct $\{f(x), |x| > t\}$ from

$\{\mathcal{R}\{f(x)\}(\tau, \phi), |\tau| > t\}$ for some $t > 0$ (note that this is the problem stated at the beginning of this section). Then (2.8) need only be applied for $|\tau| > t > 0$. The worst case of evaluating $P_m(\tau/|x|)$ in (2.8) is $m = M$, $\tau = T$ and $|x| = t$, if M is large and t is small relative to T, then $P_m(\tau/|x|) \simeq (1/2)(2T/t)^M < \infty$. Note that even this regularized problem is very poorly conditioned.

This regularization is significant to the results of this paper, since this regularized exterior Radon transform problem (with $t = 1$) will be applied to causal inversion of spherical means in subsection 3.4 below.

2.3. *Dual filtered backprojection algorithm for the entire spherical mean inverse problem*

The transforms $\mathcal{R}$ and $\mathcal{B}$ are related by [7]

$$f(x) = C_n \mathcal{B}_{(t,e)\to(|x|,e_x)} \left\{ \left(\mathcal{H} \frac{d}{dt} \right)^{n-1} \mathcal{R}_{(|x|,e_x)\to(t,e)} \{f(x)\} \right\} \tag{2.9}$$

where $\mathcal{H}$ denotes the Hilbert transform $\mathcal{H}\{f(t)\} = f(t) * \frac{1}{\pi t}$ and $C_n = (2\pi)^{1-n}$ (recall that, in the definition (2.2) of $\mathcal{B}$, S is the unit half-sphere). Discretizing this leads to the well-known technique of 'filtered backprojection' and it immediately suggests the following algorithm for inverting (1.1).

For the spherical mean problem, both $f(x)$ and its spherical means $\hat{f}(t, e)$ are continuous functions. Thus we may interchange $(|x|, e_x)$ and (t, e) in (2.9), and replace $f(x)$ with $\hat{f}(t, e)$. Then, given the spherical means $\{\hat{f}(t, e)\}$ defined in (1.1) for all (t, e), $f(x)$ may be recovered from $\{\hat{f}(t, e)\}$ using

$$\begin{aligned} f(|x|, e_x) &= \frac{1}{|x|^{n-2}} \mathcal{B}^{-1}_{(t,e)\to(|x|,e_x)} \left\{ \frac{\hat{f}(t, e)}{t} \right\} \\ &= \left(\frac{C_n}{|x|^{n-2}} \right) \left(\mathcal{H} \frac{d}{d|x|} \right)^{n-1} \mathcal{R}_{(t,e)\to(|x|,e_x)} \left\{ \frac{\hat{f}(t, e)}{t} \right\}. \end{aligned} \tag{2.10}$$

In words: Since the composition of the Radon transform, the derivative-Hilbert transform and backprojection is the identity, the inverse of the last operator is the composition of the first two operators.

2.4. *Discussion*

The above algorithm has two major advantages. First, it is relatively simple, in that it requires only a forward Radon transform and a derivative-Hilbert transform operation; the latter can be implemented in the same way that it is implemented in filtered backprojection [7]. Second, the entire spherical mean inverse problem, which this algorithm solves, is relatively well-conditioned compared to the causal spherical mean inverse problem considered in section 3.

However, this algorithm also has a major disadvantage, in that it requires knowledge of $\{\hat{f}(t, e)\}$ for all t in order to recover $f(x)$ for any x. This follows since the Radon transform $\mathcal{R}_{(t,e)\to(|x|,e_x)}$ requires $\{\hat{f}(t, e)\}$ for all t (in particular, for arbitrarily large t). Note that even if $f(x)$ has compact support, its spherical means $\{\hat{f}(t, e)\}$ will not have compact support, since even for arbitrarily large t the sphere $|x| = te - e_x$ will intersect the support of $f(x)$. Thus $\{\hat{f}(t, e)\}$ cannot be windowed arbitrarily to zero for large t.

Applying this procedure to the diffraction tomography problem of subsection 1.3, we have the amusing result that a wide-band diffraction tomography problem may be solved by Radon transforming the temporal Fourier transform of measurements at a single point, and then applying the derivative-Hilbert transform operator to obtain the scattering potential in the spatial Fourier domain! However, the data must be truly wide-band ($0 < k < \infty$), for the reason noted above.

3. Causal inversion of spherical means

Since the spherical mean is a causal operator, it should be possible to causally invert it. In order to take advantage of this causality, recall that we defined the causal inverse spherical mean problem as follows. Let $T > 0$. Given the values $\{\hat{f}(\tau, e), 0 \leqslant \tau \leqslant T\}$, determine $\{f(x), 0 \leqslant |x| \leqslant T\}$.

We present two different algorithms for solving this problem. One algorithm uses an invariant imbedding approach. The other uses geometric inversion to transform the problem into the exterior Radon transform problem. The second algorithm can also be used to solve the entire spherical mean inverse problem (see subsection 3.6); this shows the versatility of the geometric inversion approach.

3.1. Invariant imbedding algorithm

From (1.1) it is clear that $\hat{f}(t, e)$ depends on $\{f(|x|, e_x), 0 \leqslant |x| \leqslant t\}$. Suppose we increment the size of the problem from t to $t + \Delta$. This introduces an additional 'shell' of data $\{\hat{f}(t + \Delta, e), e \in S\}$, and an additional 'shell' of unknowns $\{f(|x| + \Delta, e_x), e_x \in S\}$. The additional data can then be used to determine the additional unknowns (in conjunction with previous data). Thus $f(x)$ can be recursively reconstructed in increasing $|x|$.

To show how this works, we specialize to the 2D problem; the extension to an arbitrary number of dimensions is straightforward. Let the radius $|x|$ of x in $f(x) = f(|x|, 0)$ and the radius t of its 2D spherical mean $\hat{f}(t, \phi)$ both be discretized to integer multiples of a (small) discretization length Δ. The portion of the circle $|x| = t\cos(\theta - \phi)$ that intersects the annulus $t < |x| < t + \Delta$ subtends an angle $2\Delta\theta = 2\cos^{-1}[t/(t + \Delta)]$. Assume that $f(x) = f(|x|, 0)$ does not vary significantly in the thin sector $\{(t, \theta) : t < |x| < t + \Delta, \phi - \Delta\theta < \theta < \phi + \Delta\theta\}$.

Now suppose we have reconstructed $\{f(x), 0 \leqslant |x| \leqslant t\}$, and we wish to extend the region in which $f(x)$ has been reconstructed from $0 \leqslant |x| \leqslant t$ to $0 \leqslant |x| \leqslant t + \Delta$. For each $0 \leqslant \phi < 2\pi$ we have

$$\begin{aligned}\hat{f}(t + \Delta, \phi) &= \int_{\phi - \pi/2}^{\phi + \pi/2} f(|x| = (t + \Delta)\cos(\theta - \phi), \theta)\, d\theta \\ &= \left\{\int_{\phi - \pi/2}^{\phi - \Delta\theta} + \int_{\phi + \Delta\theta}^{\phi + \pi/2}\right\} f(|x| = (t + \Delta)\cos(\theta - \phi), \theta)\, d\theta \\ &\quad + \int_{\phi - \Delta\theta}^{\phi + \Delta\theta} f(|x| = (t + \Delta)\cos(\theta - \phi), \theta)\, d\theta \\ &= \left\{\int_{\phi - \pi/2}^{\phi - \Delta\theta} + \int_{\phi + \Delta\theta}^{\phi + \pi/2}\right\} f(|x| = (t + \Delta)\cos(\theta - \phi), \theta) d\theta \\ &\quad + f(|x| = t + \Delta, \phi) 2\Delta\theta. \end{aligned} \tag{3.1}$$

The first integral in the last equality depends only on values of $f(x)$ for which $0 \leqslant |x| \leqslant t$. Since these values of $f(x)$ have already been reconstructed, the first integral may be computed from them. The 2D spherical mean $\hat{f}(t + \Delta, \phi)$ is known, so (3.1) immediately determines $f(|x| = t + \Delta, \phi)$. Using (3.1) in this way for each $0 \leqslant \phi < 2\pi$, we have extended the region in which $f(x)$ has been reconstructed from $0 \leqslant |x| \leqslant t$ to $0 \leqslant |x| \leqslant t + \Delta$.

We can then use $\hat{f}(t + 2\Delta, \phi)$ to extend the region in which $f(x)$ has been reconstructed from $0 \leqslant |x| \leqslant t + \Delta$ to $0 \leqslant |x| \leqslant t + 2\Delta$. Continuing in this way, we can reconstruct $\{f(x), 0 \leqslant |x| \leqslant T\}$ from $\{\hat{f}(\tau, e), 0 \leqslant \tau \leqslant T\}$, provided that we already know $f(x)$ in a small circle centred at the origin, namely $\{f(x), 0 \leqslant |x| \leqslant \epsilon\}$. This is necessary to initialize the above recursion. In higher dimensions, the arc $\phi - \Delta\theta < \theta < \phi + \Delta\theta$ becomes a cone $|e \cdot e_x| \leqslant t/(t + \Delta)$; if Δ is small, this cone will also be small.

3.2. Numerical example of invariant imbedding algorithm

We demonstrate the above procedure with a simple example, which is meant to be illustrative only. Consider the circularly symmetric 'mountain' function $f(|x|, 0) = e^{-|x|}, |x| \geqslant 0$ in $\mathcal{R}^2$. Its 2D spherical means (which are actually circular means) are also circularly symmetric, and a little algebra shows them to be

$$\begin{aligned}\hat{f}(t, \phi) = \hat{f}(t, 0) &= \int_{-\pi/2}^{\pi/2} te^{-t\cos\theta}\, d\theta = 2t\int_0^{\pi/2} e^{-t\cos\theta}\, d\theta \\ &= \pi t(I_0(t) - L_0(t)) \end{aligned} \tag{3.2}$$

where $I_0(t)$ is the modified Bessel function of the first kind of order zero and $L_0(t)$ is the modified Struve function of order zero. The integral definitions of $I_0(t)$ and $L_0(t)$, given in (9.6.16) and (12.2.2) of [8], respectively, were used to derive (3.2).

Before proceeding with the example, two comments are in order, based on the observation ((12.2.6) of [8]) that $\pi t(I_0(t) - L_0(t)) \simeq 2$ for large t. Firstly, even though $f(x)$ decays exponentially to zero with increasing $|x|$, its spherical means do not decay to zero at all. This is an example of how spherical means should not be windowed to zero for large t. Secondly, in the limit $t \to \infty$, the circle $|x| = t\cos\theta$ becomes a line through the origin, and

$$\hat{f}(t, \phi) \simeq \int_{-\infty}^{\infty} e^{-|r|}\, dr = 2$$

in agreement with the asymptotic behaviour of $\pi t(I_0(t) - L_0(t))$.

We chose $T = 1$ and $\Delta = 1/N$. The spherical means (3.2) were computed using Simpson's rule for numerical integration at $N + 1$ points in the interval $0 \leqslant \theta \leqslant \pi/2$. The imbedding equation (3.1) was implemented numerically by breaking up the integral over previously-reconstructed values of $f(x)$ into integrals over the annuli $\{i\Delta < |x| < (i + 1)\Delta, 1 \leqslant i \leqslant N\}$; in each annulus $f(x)$ was assumed to be constant. This yielded the recursive equation

$$\begin{aligned} f((n + 1)\Delta) &= \left(\hat{f}((n + 1)\Delta) - 2\sum_{i=1}^{n}\left[\cos^{-1}\left(\frac{i - 1}{n + 1}\right) - \cos^{-1}\left(\frac{i}{n + 1}\right)\right] f(i\Delta) i\Delta\right) \\ &\quad \times \left[2(n + 1)\Delta\cos^{-1}\left(\frac{n}{n + 1}\right)\right]^{-1} \end{aligned} \tag{3.3}$$

valid for $0 \leqslant n \leqslant N$; note $t = n\Delta$ and $T = N\Delta$.

Numerical results of running this algorithm can be summarized as follows:

(1) For $\Delta = 0.01$ (i.e. $N = 100$), the algorithm worked very well. The maximum difference between the actual and reconstructed values of $f(x)$ was 0.0136 at $|x| = 1$; a more typical difference was 0.0079 at $|x| = 0.5$. No plot is given, as the actual and reconstructed values cannot be distinguished on it.

(2) Recall that $f(x)$ must be already known inside a circle of radius ϵ, to initialize the invariant imbedding recursion. For this example, the minimum value of ϵ was found to be Δ; specifying $f(x)$ at locations other than the origin did not improve the reconstruction.

(3) Decreasing Δ had little effect on the accuracy of the reconstruction. For example, $\Delta = 0.005$ ($N = 200$) resulted in differences of 0.0109 at $|x| = 0.5$ and 0.0154 at $|x| = 1$. The improved accuracy in modelling the continuous function $f(x)$ seems to be offset by roundoff error; note that division by $2(n+1)\Delta \cos^{-1}(n/(n+1))$ is required in (3.3). Roundoff error can be ameliorated somewhat by precomputing the coefficient of $f(i\Delta)$ as

$$\left(\frac{i}{n+1}\right)\left[\cos^{-1}\left(\frac{i-1}{n+1}\right) - \cos^{-1}\left(\frac{i}{n+1}\right)\right]\left[\cos^{-1}\left(\frac{n}{n+1}\right)\right]^{-1}$$

but roundoff error may still be a problem with this algorithm.

If $f(x)$ is not circularly symmetric, difficulties will arise near the origin, since $\Delta\theta$ is no longer small and the assumption that $f(x)$ is constant in a wedge subtending angle $\Delta\theta$ is no longer valid. There are two ways around this difficulty. One way is to make ϵ, the radius of the circle in which $f(x)$ is assumed to be already known, large enough so that $\Delta\theta$ at radius ϵ is small enough for $f(x)$ to be assumed constant in the wedge subtending angle $\Delta\theta$. The recursion is then initialized at radius $|x| = \epsilon + \Delta$. The other way is explained next.

3.3. Invariant imbedding algorithm for bandlimited functions

If $f(x)$ is assumed to be angularly bandlimited, as discussed in subsection 2.2, the following changes can be made in the invariant imbedding algorithm. Let $F_m(|x|)$ defined in (2.7*a*) be zero for $|m| > M$, for some M. Then both $f(|x|, \theta)$ and $\hat{f}(t, \phi)$ can be sampled in their angular variables at integer multiples of $2\pi/(2M+1)$ with no loss of information, since the samples may be interpolated to give the original functions. Suppose the data consist of the angularly-sampled 2D spherical means $\{\hat{f}(t, \phi_k), \phi_k = 2\pi k/(2M+1), 0 \leqslant k \leqslant 2M\}$. Also suppose $f(|x|, \theta)$ can be assumed to be piecewise constant in $|x|$, changing only at integer multiples of Δ. Writing (3.1) for $\phi = \phi_k, 0 \leqslant k \leqslant 2M$, substituting (2.6) in (3.1), and breaking up the integral in (3.1) into integrals over the annuli $\{n\Delta \leqslant |x| \leqslant (n+1)\Delta, 0 \leqslant n \leqslant N\}$ where $t = N\Delta$ yields the $2M+1$ simultaneous linear equations in $2M+1$ unknowns $F_m((N+1)\Delta)$

$$\hat{f}((N+1)\Delta, \phi_k) - \sum_{n=1}^{N}\sum_{m=-M}^{M} A_{m,n,k} F_m(n\Delta) = \sum_{m=-M}^{M} A_{m,N+1,k} F_m((N+1)\Delta) \qquad 0 \leqslant k \leqslant 2M \tag{3.4a}$$

$$A_{m,n,k} = \frac{2n\Delta}{m} e^{im\phi_k}\left[\sin\left(m\cos^{-1}\left(\frac{n-1}{N+1}\right)\right) - \sin\left(m\cos^{-1}\left(\frac{n}{N+1}\right)\right)\right] \tag{3.4b}$$

The linear system of equations (3.4) can be solved for the $F_m((N+1)\Delta)$, which can then be inserted into (2.6*a*) to give the reconstructed $f(|x| = t + \Delta, \theta)$. Note that setting $M = 0$ in (3.4) and using the relation $\lim_{m\to 0} \sin(mx)/m \simeq x$ yields (3.3). However, the system of equations (3.4) becomes poorly-conditioned for large m; this is not surprising, for reasons explained in subsection 3.5 below.

3.4. Geometric inversion algorithm for the causal spherical mean inverse problem

Another approach to solving the causal spherical mean inverse problem is as follows. Given spherical means $\{\hat{f}(t, e), t < T\}$ of $\{f(x), |x| < T\}$, the values $\{f(x), |x| < T\}$ may be recovered from $\{\hat{f}(t, e), t < T\}$ as follows:

(1) Define $\hat{g}(t, e) = T^{n-1}\hat{f}(t, e)$ for $t < T$.
(2) Define $\hat{g}'(t, e) = \hat{g}(T/t, e)$ for $t > 1$.
(3) Compute $g'(|x|, e_x) = \mathcal{R}^{-1}_{(t,e)\to(|x|,e_x)}\{\hat{g}'(t, e)\}$ for $t > 1$ and $|x| > 1$.
(4) Define $g(|x|, e_x) = g'(T/|x|, e_x)$ for $|x| < T$.
(5) Then $f(x) = g(|x|, e_x)/|x|^{2(n-1)}$ for $|x| < T$.

Note here that primes denote quantities that have been inverted across a sphere, and hats denote quantities that are transforms of unhatted quantities.

To prove this result we need the following lemma:

Lemma. Let $g(x)$ be any continuous function on $\mathcal{R}^n$, and consider the inversion of the Radon transform of its inversion $\{\mathcal{R}_{(|x|,e_x)\to(t,e)}\{g(T/|x|, e_x)\}_{t\to T/t}\}$. Then

$$\{\mathcal{R}_{(|x|,e_x)\to(t,e)}\{g(T/|x|, e_x)\}_{t\to T/t}\} = tT^{n-1}\mathcal{B}_{(|x|,e_x)\to(t,e)}\{g(|x|, e_x)/|x|^n\} \tag{3.5}$$

Proof. We have

$$\begin{aligned}
&\{\mathcal{R}_{(|x|,e_x)\to(t,e)}\{g(T/|x|, e_x)\}_{t\to T/t}\} \\
&\quad = \int_0^\infty \int_S g(T/|x|, e_x)\delta(T/t - |x|e\cdot e_x)|x|^{n-1}\,de_x\,d|x| \\
&\quad = \int_0^\infty \int_S g(|x|, e_x)\delta\left(\frac{T}{t} - \frac{T}{|x|}(e\cdot e_x)\right)\frac{T^n}{|x|^{n+1}}\,de_x\,d|x| \\
&\quad = \int_0^\infty \int_S g(|x|, e_x)\delta(|x| - te\cdot e_x)\frac{tT^{n-1}}{|x|^n}\,de_x\,d|x| \\
&\quad = tT^{n-1}\int_S g(|x| = te\cdot e_x, e_x)/|x|^n\,de_x \\
&\quad = tT^{n-1}\mathcal{B}_{(|x|,e_x)\to(t,e)}\{g(|x|, e_x)/|x|^n\}.
\end{aligned} \tag{3.6}$$

In (3.6) we changed variables from $|x|$ to $T/|x|$ and used the scale property $\delta\left(\frac{yT}{|x|t}\right) = \frac{|x|t}{T}\delta(y)$ of the impulse.

The algorithm follows immediately from the lemma and (2.4) by setting

$$f(x) = g(|x|, e_x)/|x|^{2(n-1)}$$
$$\hat{f}(t, e)T^{n-1} = \{\mathcal{R}_{(|x|,e_x)\to(t,e)}\{g(T/|x|, e_x)\}_{t\to T/t}\}.$$

Note that step 3 is the regularized exterior Radon transform problem (we only require $g'(|x|, e_x)$ for $|x| > 1$), which can be solved as discussed in subsection 2.2.

3.5. Comments

The form of the algorithm in subsection 3.4 is not surprising: as shown in subsection 1.4, geometrically inverting a plane across the sphere of radius T results in a sphere which passes through the origin, and vice versa. This explains why the inversions $t \to T/t$ and $|x| \to T/|x|$ transform a problem involving integrals over spheres into a problem involving integrals over planes.

Note that the inverse Radon transform in step 3 could be performed by a filtered backprojection operation (2.9), if not for the restrictions on $|x|$ and t. This is not surprising; previous work on inversion of spherical means has generally involved backprojection over spheres, or equivalent operations [1, 2]. The difference between previous work and the above algorithm is that the causality restrictions imposed on the problem preclude the use of backprojection to compute $\mathcal{R}^{-1}$ in step (3).

The inverse Radon transform in step (3) must be performed anti-causally, computing $\{g'(|x|, e_x), |x| > 1\}$ from $\{\hat{g}'(t, e), t > 1\}$; this is the regularized exterior Radon transform problem described in subsection 2.2. Note that since we only require $g'(|x|, e_x)$ for $|x| > 1$ (i.e. $t = 1$ in section 2.2), the problem is not ill-conditioned; however, it will be poorly conditioned (see the end of subsection 2.2).

We also require that $g'(|x|, e_x)$ be known to equal zero for $|x| > R$ for some arbitrarily large R; this is equivalent to constraining $f(|x|, e_x) = 0$ for x inside the small sphere $|x| < \epsilon$, where $\epsilon = T/R$ (this also avoids difficulties with $g(|x|, e_x)/|x|^n$ increasing without bound at the origin). Thus once again we see that knowledge of $f(x)$ inside a small sphere is necessary for $\{f(x), |x| < T\}$ to be uniquely determined from $\{\hat{f}(t, e), t < T\}$.

Specifically, suppose we wish to reconstruct $\{f(x), |x| < T\}$ from its 2D spherical means $\{\hat{f}(t, \phi), t < T\}$, and that $f(x)$ is known to be zero for $|x| < \epsilon$. The worst case of evaluating the Chebyschev polynomial in (2.8) is then

$$P_M(R/1) = P_M(T/\epsilon) \simeq (1/2)(2T/\epsilon)^M$$

for large M and T/ϵ. Hence the larger the radius ϵ of the circle of already-known values of $f(x)$, relative to the radius T of the circle inside which the inverse spherical mean problem is defined, the better the conditioning of the inverse problem will be. From a purely geometric viewpoint, the significance of the ratio T/ϵ, as opposed to the individual values of T and ϵ, is clear.

In fact, it is only necessary that $f(x)$ be known inside the small sphere $|x| < \epsilon$; it need not be zero inside this small sphere. To see this, let $f(x) = f_{ext}(x) + f_{int}(x)$, where $f_{ext}(x) = 0$ for $|x| < \epsilon$, and $f_{int}(x) = 0$ for $|x| > \epsilon$ (outside the small sphere) and takes on the known values of $f(x)$ for $|x| < \epsilon$. Since $f_{int}(x)$ is known completely, its spherical means may be computed and subtracted from the given $\hat{f}(t, e)$, resulting in the spherical means $\hat{f}_{ext}(t, e)$ of $f_{ext}(x)$. The above algorithm is then used to determine $f_{ext}(x)$ from $\hat{f}_{ext}(t, e)$, and then $f(x)$ is completely determined.

3.6. Geometric inversion algorithm for the entire spherical mean inverse problem

Recall that the entire spherical mean inverse problem is to reconstruct $f(x)$ from $\{\hat{f}(t, e)\}$ for all t and e. The procedure given in subsection 3.4 for the causal problem can also be applied to the entire problem, simply drop the modifying inequalities $t < T$, $t > 1$, etc since the lemma did not require these. Note that a sphere of very large radius t geometrically inverts to a plane passing very close to the origin;

integrals over these planes are not used in the exterior Radon transform problem, but they are used in the complete Radon transform problem of reconstructing $f(x)$ from all of its integrals over hyperplanes.

Thus the entire spherical mean inverse problem is equivalent, through geometric inversion, to the complete Radon transform problem. This is significant, since the complete Radon transform problem is relatively well-conditioned, and it may be solved quite readily using filtered back-projection [7]. In contrast, the exterior Radon transform problem, to which the causal spherical mean inverse problem is equivalent through geometric inversion, is relatively poorly conditioned, and it is generally considered to be impractical using real-world data.

Also note that, unlike the dual filtered backprojection algorithm of subsection 2.3, we do not require the data $\hat{f}(t, e)$ for all t to compute $f(x)$ for any x. Indeed, we do not require $\hat{f}(t, e)$ for $t > 1/\epsilon$ if $f(x)$ is known for $|x| < \epsilon$ —we merely proceed as in subsection 3.5 above. This algorithm appears to be the most useful one of those proposed in the current paper, provided the necessary data are available.

4. Nullspace of spherical means

4.1. Characterization of nullspace

Without the restriction that $f(x)$ is known inside a small sphere, the modified spherical mean inverse problem does not have a unique solution. That is, $\mathcal{B}$ has a non-empty nullspace $\mathcal{N}$, where

$$\mathcal{N} = \{f : (\mathcal{B}f)(t, e) = 0 \quad \text{for } t < T\}$$

(recall the problem is only defined for $|x| < T$ and $t < T$). The non-uniqueness then follows since it is possible for two different functions $f_1(x)$ and $f_2(x)$ to have the same set of spherical means for $t < T$; all that is needed is $f_1(x) - f_2(x) \in \mathcal{N}$. We now investigate the nullspace $\mathcal{N}$ of $\mathcal{B}$.

Let $f_{ext}(x)$ be any function such that $f_{ext}(x) = 0$ for $|x| < T$, i.e. $f_{ext}(x)$ is a 'hollow' function. Then $(\mathcal{H}\frac{d}{dt})^{n-1}\mathcal{R}_{(|x|,e_x)\to(t,e)}\{f_{ext}(x)\} \in \mathcal{N}$ by (2.9). Hence we can construct members of $\mathcal{N}$ by taking any hollow function and applying the operator $(\mathcal{H}\frac{d}{dt})^{n-1}\mathcal{R}$ to it. This might seem like a large class of non-zero functions, but the following theorem shows that this class is not as large as it might seem.

Theorem. Let $f_{ext}(x) = 0$ for $|x| < T$ so that $f_{ext}(x)$ is a hollow function. Compute (compare to (2.10))

$$\check{f}_{ext}(t, e) = \mathcal{B}^{-1}_{(|x|,e_x)\to(t,e)}\{f_{ext}(x)\} = \left(\mathcal{H}\frac{d}{dt}\right)^{n-1}\mathcal{R}_{(|x|,e_x)\to(t,e)}\{f_{ext}(x)\} \tag{4.1a}$$

$$\check{f}_{ext}(t, e) = \mathcal{B}^{-1}_{(|x|,e_x)\to(t,e)}\{f_{ext}(x)\} \in \mathcal{N} \tag{4.1b}$$

Let $\epsilon > 0$ be arbitrarily small. Then if $\check{f}_{ext}(t, e) = 0$ for $t < \epsilon$, then $\check{f}_{ext}(t, e) = 0$ for $t < T$, so that $\check{f}_{ext}(t, e)$ is a trivial member of $\mathcal{N}$ (zero everywhere in the region of interest $t < T$). The significance of this theorem is that although $f_{ext}(x)$ is zero for $|x| < T$, $(\mathcal{H}\frac{d}{dt})^{n-1}\mathcal{R}\{f_{ext}(x)\}$ must be non-zero near the origin $t = 0$ for it to be a non-trivial member of $\mathcal{N}$. Hence some choices of $f_{ext}(x)$ will give rise to trivial members of $\mathcal{N}$, so $\mathcal{N}$ is not as large as it seems.

Proof 1. Using geometric reflection as in subsection 3.3, this result follows immediately from the uniqueness result of the exterior Radon transform (see subsection 2.2). □

Proof 2. Exchange $(|x|, e_x)$ and (t, e), and apply the lemma with

$$g(|x|, e_x) = |x|^n \check{f}_{ext}(|x|, e_x)$$
$$tT^{n-1} f_{ext}(t, e) = \{\mathcal{R}_{(|x|,e_x)\to(t,e)}\{g(T/|x|, e_x)\}_{t\to T/t}\}.$$

Again the result follows from uniqueness of the exterior Radon transform. □

Proof 3. From the definition (2.1) of the Radon transform and the fact that $f_{ext}(|x|, e_x) = 0$ for $|x| < T$

$$\check{f}_{ext}(t, e) = \int_T^\infty \int_S f_{ext}(|x|, e_x) \left(\mathcal{H}\frac{\mathrm{d}}{\mathrm{d}t}\right)^{n-1} \delta(t - |x|e \cdot e_x)|x|^{n-1} \mathrm{d}e_x \, \mathrm{d}|x|. \tag{4.2}$$

Since $t < T < |x|$ always, the integrand will always have two singularities at $t/|x| = e \cdot e_x$. Since the integral is zero for all $t < \epsilon$, $f_{ext}(|x|, e_x)$ must equal zero at these two singularities, so we have $f_{ext}(|x| = t/(e \cdot e_x), e_x) = 0$ for all $t < \epsilon$ and all e. Hence we may choose e sufficiently close to orthogonal to e_x to make $e \cdot e_x = \epsilon/t$ for any $t < T$. Then, since $f(\epsilon/(e \cdot e_x), e_x) = 0$, we have $f(t, e_x) = 0$ for any t such that there are singularities, i.e. for any t such that $t < T < |x|$.

4.2. Example

It may seem strange that while $\mathcal{R}\{f_{ext}(x)\}$ is clearly non-zero for $t < T$, its filtration $\check{f}_{ext}(t, e) = (\mathcal{H}\frac{\mathrm{d}}{\mathrm{d}t})^{n-1}\mathcal{R}\{f_{ext}(x)\}$ may be zero for $t < T$. To illustrate this, let $T = 1$ and consider the 'hollow ball' (in $\mathcal{R}^3$) function $f_{ext}(x) = 1$ if $1 < |x| < 2$ and $= 0$ otherwise. Then

$$\mathcal{R}_{(|x|,e_x)\to(t,e)}\{f_{ext}(x)\} = \begin{cases} 0 & \text{if } t > 2 \\ \pi(4 - t^2) & \text{if } 1 < t < 2 \\ 3\pi & \text{if } 0 < t < 1 \end{cases} \tag{4.3}$$

so that although $\mathcal{R}\{f_{ext}(x)\} \neq 0$ for $t < 1$, $\check{f}_{ext}(t) = \frac{\mathrm{d}^2}{\mathrm{d}t^2}\mathcal{R}\{f_{ext}(x)\} = 0$ for $t < 1$.

5. Conclusion

We have presented several procedures for reconstructing a function $f(|x|, e_x)$ from its spherical means $\hat{f}(t, e)$ on spheres that pass through the origin. If we are given the data $\hat{f}(t, e)$ for all sphere diameters t (the entire problem), $f(|x|, e_x)$ may be reconstructed directly using the Radon transform and a derivative-Hilbert transform (see subsection 2.3). If we are given only $\{\hat{f}(t, e), 0 \leqslant t \leqslant T\}$ (the causal problem), we may reconstruct $\{f(|x|, e_x), 0 \leqslant |x| \leqslant T\}$ using invariant imbedding; a simple numerical example was given, and a variation for bandlimited functions $f(|x|, e_x)$ was also given (see subsections 3.1–3.3).

Both the entire and causal problems may also be solved by inverting the data across the sphere of radius T, performing a complete inverse Radon transform (for the entire problem) or an anti-causal or exterior inverse Radon transform (for the causal problem), and then inverting the result about the sphere of radius T. These geometric inversions map the spheres over which the function is integrated into planes, thus mapping the spherical mean transform problem into a Radon transform problem. Since the Radon transform problem is well understood, this geometric inversion approach would seem to be preferable. Note that the complete Radon transform problem is relatively well conditioned, while the exterior Radon transform problem is relatively poorly conditioned; similar remarks seem to apply to the entire and causal spherical mean problems, respectively.

The problem was shown to have a unique solution, provided that the function is specified on an arbitrarily small sphere centred on the origin; the nullspace of the spherical mean operator missing this specification was also investigated. One interesting possibility for further research would be to solve the exterior Radon transform problem by reflecting it into a spherical mean problem, and then applying the invariant imbedding algorithm to solve this problem. Note that invariant imbedding cannot be applied directly to the exterior Radon transform problem, since it would operate on the unbounded domain $|x| > t$.

Acknowledgments

The author would like to thank one of the reviewers for a careful reading of the first version of this paper. This research was supported by the Office of Naval Research under grant N00014-90-J-1897.

References

[1] John F 1955 *Plane Waves and Spherical Means* (New York: Wiley)
[2] Fawcett J A 1985 Inversion of N dimensional spherical averages *SIAM J. Appl. Math.* **45** 336–41
[3] Cohen J and Bleistein F 1979 Velocity inversion procedure for acoustic waves *Geophysics* **44** 1077–85
[4] Norton S J and Linder M 1981 Ultrasonic reflectivity imaging in three dimensions: exact inverse scattering solutions for plane, cylindrical, and spherical apertures *IEEE Trans. Biomed. Eng.* **BME-28** 202–20
[5] Devaney A J 1984 Geophysical diffraction tomography *IEEE Trans. Geosci.* **GE-22** 3–13
[6] Cormack A M 1964 Representation of a function by its line integrals, with some radiological applications II *J. Appl. Phys.* **35** 2908–13
[7] Deans S R 1983 *The Radon Transform and Some of its Applications* (New York: Wiley)
[8] Abramowitz M and Stegun I 1965 *Handbook of Mathematical Functions* (New York: Dover)

APPENDIX E1

J. Frolik and A.E. Yagle, "Forward and Inverse Scattering for Discrete Layered Lossy and Absorbing Media," submitted to SIAM J. Appl. Math, June 1993.

A complete theory for the 1-D forward and inverse scattering problems for discrete layered lossy media is presented. By a "complete theory," we mean systems of equations that are discrete counterparts to integral equations, and discrete fast algorithms that solve these systems of equations. Applications to discrete lossy transmission lines, and to electromagnetic wave propagation in absorbing dielectrics, are made, and numerical examples presented.

FORWARD AND INVERSE SCATTERING FOR DISCRETE LAYERED LOSSY AND ABSORBING MEDIA

Jeffrey L. Frolik and Andrew E. Yagle

Dept. of Electrical Engineering and Computer Science

The University of Michigan, Ann Arbor, MI 48109-2122

May 1993

Abstract. A complete digital signal processing (DSP) theory is developed for forward and inverse scattering in discrete (piecewise-constant) layered lossy systems, generalizing previous work for discrete lossless systems and continuous lossy systems. This work is motivated by radar reflections from stratified dielectrics and interchip communication modeled by lossy transmission lines. The DSP formulation allows exact solutions, without discretization approximations, and includes all multiple reflections, transmission scattering losses, and absorption effects. For the forward problem, discrete matrix Green's functions are derived for the lossy medium, as are fast algorithms for computing impulse reflection and transmission responses for the medium. For the inverse problem, asymmetric Toeplitz systems of equations which function as discrete counterparts to integral equations are derived, as are fast algorithms for reconstructing the medium from its impulse transmission and reflection responses. Data sufficiency and feasibility are discussed. Finally, these results are applied to the LCRG transmission line and layered dielectric medium problems, dispersion effects are discussed, and a numerical example is presented.

Keywords. Inverse scattering, lossy media, layered dielectrics, lossy transmission lines.

AMS (MOS) subject classification. 34A55, 35P25, 35R30, 39A12

1. Introduction. Inverse scattering problems have application in numerous fields, including mathematics, geophysics, physics, and electrical engineering. The study of inverse scattering for electrical engineers is motivated by the mathematical similarity of problems such as the design of digital filters in cascade form, the derivation of fast lattice-form linear least-squares prediction error filters, synthesis of transmission lines, and wave propagation in layered dielectric media, etc. In each case, the goal is to compute system parameters (e.g., PARCOR coefficients, dielectric constants, wave speeds, etc.) from the response to an impulsive excitation measured at the system boundaries. This response is known as the *scattering data.*

Since many of these systems are discrete in nature (e.g., digital filters, discrete layered dielectric media, etc.), they can be modeled as *discrete-time* systems. The discrete version of the inverse scattering problem has been applied to lossless transmission lines [1], Schrödinger's equation [2], elastic waves in layered media [3], and electromagnetic scattering [4]. However, all of these were lossless inverse scattering problems.

In this paper, we will extend previous work [1], which has dealt entirely with *discrete lossless* systems, to the case of *discrete lossy* or *absorbing* systems. In particular, we will discuss and link the following topics: (1) scattering (forward problem); (2) reconstruction (inverse problem); (3) discrete matrix Green's functions; (4) discrete asymmetric Toeplitz systems; (5) asymmetric Levinson and Schur algorithms; and (6) applications to lossy transmission lines and electromagnetic wave propagation in layered dielectrics.

The advantages of the discrete formulation over the continuous formulation used in [5] and [6] are three-fold. First, most practical problems of interest are in fact best modeled as discrete (piecewise-constant), rather than continuous, systems. The present work was motivated by radioglaciology (radar propagation through glaciers [7]); other applications include radar reflections from stratified dielectrics [8], interchip communication modeled by lossy transmission lines [9], and design of optical waveguide gratings [10]. Second, our digital signal processing (DSP) formulation explicitly shows the connection between scattering from a discrete *lossy* system and using the Levinson and Schur algorithms for *asymmetric* Toeplitz systems of equations. This

generalizes previous work [1] connecting scattering from a *lossless* medium and the Levinson and Schur algorithms for *symmetric* Toeplitz systems of equations. It also permits the use of such familiar DSP tools as the z-transform and discrete Fourier transform, which can be computed easily and exactly, rather than the continuous Fourier transform, which must always be approximated and which may introduce aliasing. Finally, important issues such as transmission losses and dispersion appear in the discrete formulation that do not appear in the continuous formulation of [5] and [6]. Since continuous equations are discretized in [5] and [6], these effects are neglected there. This makes our explicitly discrete approach more realistic, both computationally and in modeling.

This paper is organized as follows. In Section 2 we define the asymmetric two-component wave system that models wave propagation in lossy media (as will be shown in Sections 5 and 6). Section 3 defines the forward scattering problem, various types of scattering data, and derives the discrete matrix Green's function for the medium. This can be used to generate scattering data from medium parameters. We also discuss sufficiency and feasibility of scattering data. Section 4 defines the inverse scattering problem, and shows how it may be solved either by solving an asymmetric Toeplitz system of equations, which functions as a discrete counterpart to an integral equation, or by running the asymmetric Schur algorithm. Section 5 shows how a discrete lossy (LCRG) transmission line can be formulated as in Section 2. Section 5 also discusses the significance of dispersion, relative to absorption. Section 6 shows how electromagnetic wave propagation in a layered dielectric can be formulated as in Section 2. Although there have been many matrix formulations of this problem, ours is the first that can be linked directly to asymmetric Toeplitz systems and already-existing fast algorithms. Section 7 presents a illustrative numerical example demonstrating the reconstruction of a lossy layered dielectric. Section 8 concludes with a summary.

2. System Definitions.

2.1. Discrete Systems. A discrete system is defined to consist of homogeneous segments (or layers), in each of which the characterizing parameters (e.g., impedance, wave speed, etc.) do not change. An example of a discrete system, a transmission line in which x is horizontal position and Z is characteristic impedance, is shown in Fig. 1.

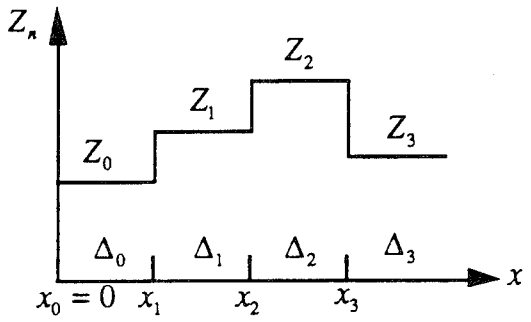


Fig. 1 Impedance profile for a discrete system

We assume that the travel time through a homogeneous layer is an integer multiple of a small time increment Δ. For example, if wave speed is constant in Fig. 1, then we assume $\Delta_n = n_n\Delta$ for some integer n_n. Note that : (1) homogeneous layers of varying thicknesses are effectively being modeled as stacks of thin, identical layers; (2) this situation can be realized by integrating and then sampling continuous-time data; and (3) this assumption is also required in [1]. Since Δ is merely a scaling factor, without loss of generality we set $\Delta = 1$ for convenience.

2.2. Discrete Asymmetric Two-Component Wave Systems. A segment of a discrete lossy system can be described by the asymmetric two-component wave system shown in Fig. 2.

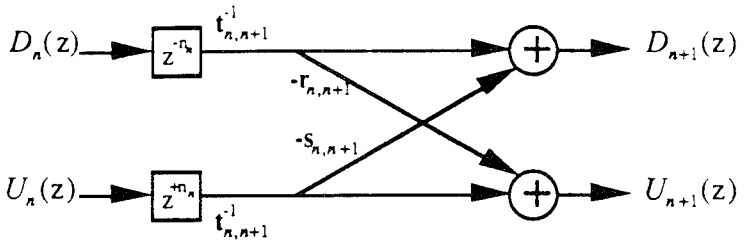


Fig. 2 Transmission diagram for a discrete lossy system

$D_n(z)$ and $U_n(z)$ are the z-transforms of the Downgoing and Upgoing waves, respectively, and are defined just to the right of the interface between the $(n-1)^{st}$ and n^{th} segments. The z-transform of a discrete-time sequence $\{...x_{-2},x_{-1},x_0,x_{+1},x_{+2}...\}$ is defined as $X(z)=Z\{x(n)\}=\sum_{i=-\infty}^{\infty}x_i z^{-i}$. The relationship between $\{D_n(z),\ U_n(z)\}$ and $\{D_{n+1}(z),\ U_{n+1}(z)\}$ is characterized by the layer propagation matrix $\{\mathbf{F}_{n,n+1}(z)\}$

$$(2.1)\quad \begin{bmatrix} D_{n+1}(z) \\ U_{n+1}(z) \end{bmatrix} = \mathbf{F}_{n,n+1}(z) \begin{bmatrix} D_n(z) \\ U_n(z) \end{bmatrix} = \frac{1}{\sqrt{1-\mathbf{r}_{n,n+1}\mathbf{s}_{n,n+1}}} \begin{bmatrix} 1 & -\mathbf{s}_{n,n+1} \\ -\mathbf{r}_{n,n+1} & 1 \end{bmatrix} \begin{bmatrix} z^{-n_n} & 0 \\ 0 & z^{n_n} \end{bmatrix} \begin{bmatrix} D_n(z) \\ U_n(z) \end{bmatrix},$$

which we write as the product of a z-independent transfer matrix $\{\Sigma_{n,n+1}\}$ having transfer coefficients $\{\mathbf{r}_{n,n+1},\ \mathbf{s}_{n,n+1}\}$ and a time-delay matrix $\{\Phi_{n,n+1}(z)\}$, so that

$$(2.2a)\quad \Sigma_{n,n+1} = \frac{1}{\sqrt{1-\mathbf{r}_{n,n+1}\mathbf{s}_{n,n+1}}} \begin{bmatrix} 1 & -\mathbf{s}_{n,n+1} \\ -\mathbf{r}_{n,n+1} & 1 \end{bmatrix}; \qquad (2.2b)\quad \Phi_{n,n+1}(z) = \begin{bmatrix} z^{-n_n} & 0 \\ 0 & z^{n_n} \end{bmatrix}.$$

The *lossy* model differs from the *lossless* model in that the transfer matrix coefficients $\{\mathbf{r}_{n,n+1},\ \mathbf{s}_{n,n+1}\}$ are *not equal*. Note that $\mathbf{t}_{n,n+1}=\sqrt{1-\mathbf{r}_{n,n+1}\mathbf{s}_{n,n+1}}$ is the transmission coefficient for the layer, for reasons that will become clear in Section 3. The model (2.1) is motivated by the applications discussed in Sections 5 and 6.

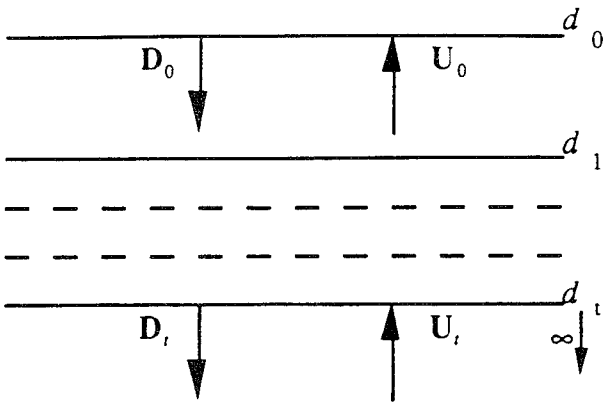


Fig. 3 Discrete multi-layer model

3. Forward Scattering.

3.1. Transition Matrix. For the discrete multi-layer medium shown in Fig. 3, the relationship between the waves in any two segments can be found by cascading the intervening layer propagation matrices $\{\mathbf{F}_{n,n+1}(z)\}$ defined in (2.1). Specifically, the waves $\{\mathbf{D}_t = D_t(z), \mathbf{U}_t = U_t(z)\}$ in the *last* layer are related to those $\{\mathbf{D}_0 = D_0(z), \mathbf{U}_0 = U_0(z)\}$ in the *first* layer by

$$\begin{bmatrix} D_t(z) \\ U_t(z) \end{bmatrix} = \prod_{i=0}^{t-1} \mathbf{F}_{i,i+1} \begin{bmatrix} D_0(z) \\ U_0(z) \end{bmatrix} = \mathbf{M}_t(z) \begin{bmatrix} D_0(z) \\ U_0(z) \end{bmatrix}, \tag{3.1}$$

where the product of the intervening layer matrices $\{\mathbf{F}_{n,n+1}(z)\}$ is the *transition* matrix $\{\mathbf{M}_t(z)\}$ for the total medium, which is a 2x2 matrix of polynomials

$$\mathbf{M}_t(z) = \begin{bmatrix} F_{11}(z) & F_{12}(z) \\ F_{21}(z) & F_{22}(z) \end{bmatrix}. \tag{3.2}$$

Note that $\mathbf{M}_t(z)$ acts as the z-transform of a discrete Green's function for the entire medium, in that it relates the waves at one end to the waves at the other end. As an aside, note that for lossless media we have $\mathbf{r}_{n,n+1} = \mathbf{s}_{n,n+1}$, and the elements of the transition matrix can easily be shown to satisfy the following relationship [1]:

(3.3) $$\mathbf{M}_t(z)=\begin{bmatrix} F_{11}(z) & F_{12}(z) \\ F_{12}(z^{-1}) & F_{11}(z^{-1}) \end{bmatrix}.$$

3.2. Reflection and Transmission Responses. We define the *forward reflection* $\{R^f(z)\}$ and *transmission* $\{T^f(z)\}$ responses as the responses of the layered medium to a downgoing impulse at the top layer (see (3.4a)). Likewise, we define the *backward reflection* $\{R^b(z)\}$ and *transmission* $\{T^b(z)\}$ responses as the responses to an upgoing impulse at the bottom layer (see (3.4b)).

(3.4a) $$\begin{bmatrix} T^f(z) \\ 0 \end{bmatrix}=\mathbf{M}_t(z)\begin{bmatrix} 1 \\ R^f(z) \end{bmatrix}$$ (3.4b) $$\begin{bmatrix} R^b(z) \\ 1 \end{bmatrix}=\mathbf{M}_t(z)\begin{bmatrix} 0 \\ T^b(z) \end{bmatrix}$$

$R^f(z)$, $T^f(z)$, $R^b(z)$, and $T^b(z)$ are known collectively as the *half-space impulse responses.* They are illustrated in Fig. 4 (note $1=Z\{\delta(n)\}$).

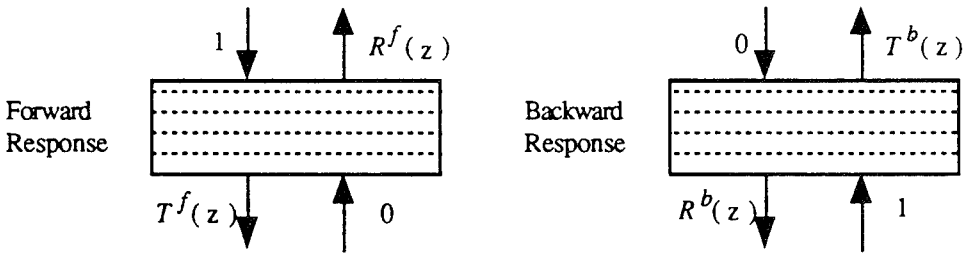


Fig. 4 Half-space impulse responses

It can be shown that the forward and backward transmission responses are identical, i.e., we have the reciprocity relation $T^f(z)=T^b(z)$. We also have

(3.5) $$DET\{\mathbf{M}_t(z)\}=F_{11}(z)F_{22}(z)-F_{12}(z)F_{21}(z)=1.$$

Note that (3.5) follows from the layer matrices $\{ \mathbf{F}_{n,n+1}(z) \}$ defined in (2.1) being unimodular, and that the product of unimodular matrices is itself unimodular. Recall that a unimodular matrix has determinant $\{ DET \}$ equal to unity, with entries functions of some variable, e.g., z.

The outgoing wave components $\{ D_t(z),\ U_0(z) \}$ can be written as a function of the incoming wave components $\{ D_0(z),\ U_t(z) \}$ using the *scattering matrix* $\{ \mathbf{S}(z) \}$

$$(3.6) \qquad \begin{bmatrix} D_t(z) \\ U_0(z) \end{bmatrix} = \begin{bmatrix} T^f(z) & R^b(z) \\ R^f(z) & T^b(z) \end{bmatrix} \begin{bmatrix} D_0(z) \\ U_t(z) \end{bmatrix} = \mathbf{S}(z) \begin{bmatrix} D_0(z) \\ U_t(z) \end{bmatrix}$$

and Fig. 2, which depicted the wave propagation (2.1), can be redrawn as the scattering diagram Fig. 5.

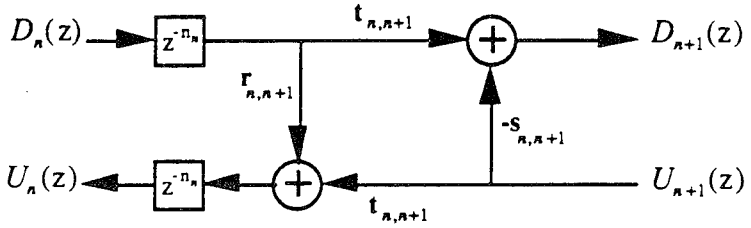


Fig. 5 Scattering diagram for discrete lossy system

The transition matrix formulation has the advantage that the wave response from one layer to the next is found simply by multiplying the intervening layer propagation matrix. This method does not hold true for the scattering formulation; however, the system input/output relationships are best described using the scattering formulation.

The reflection and transmission responses $R^b(z)$ and $T^b(z)$ can easily be computed, as can be seen by comparing Fig. 3 and Fig. 4. Given $\{ \mathbf{r}_{n,n+1} \}$ and $\{ \mathbf{s}_{n,n+1} \}$, compute $\mathbf{F}_{n,n+1}(z)$ for each n using (2.1), and $\mathbf{M}_t(z)$ using (3.1). From (3.2) and (3.4b), it can be seen that

$$(3.7a) \qquad R^b(z) = \frac{F_{12}(z)}{F_{22}(z)}; \qquad (3.7b) \qquad T^b(z) = \frac{1}{F_{22}(z)}.$$

$R^f(z)$ and $T^f(z)$ can be computed similarly from the elements of $\mathbf{M}_i^{-1}(z)$. Computing $\mathbf{M}_i^{-1}(z)$ in closed form, we have

(3.7c) $$R^f(z) = \frac{-F_{21}(z)}{F_{22}(z)};$$ (3.7d) $$T^f(z) = F_{11}(z) - \frac{F_{12}(z)F_{21}(z)}{F_{22}(z)}.$$

3.3. Adjoint System. In previous work [1], it was shown that when the system is lossless, and therefore the transfer coefficients are equal $\{ \mathbf{r}_{n,n+1} = \mathbf{s}_{n,n+1} = R_{n,n+1} \}$, the reflection coefficients $\{ R_{n,n+1} \}$ at each boundary can be found by measuring the *one-sided* impulse reflection response $\{ R^f(z) \}$ of the system. On the other hand, when the system is lossy, and the transfer coefficients are not equal $\{ \mathbf{r}_{n,n+1} \neq \mathbf{s}_{n,n+1} \neq R_{n,n+1} \}$, additional information is clearly needed.

The additional information can be found in the impulse reflection response of the *adjoint* system defined in (3.8) below. The adjoint system does not physically exist, and thus its impulse response cannot be directly measured. However, we will show shortly that by measuring the *two-sided* impulse response of the system, the adjoint impulse reflection response can be calculated.

We start by characterizing the adjoint system by a forward layer propagation matrix $\{ \tilde{\mathbf{F}}_{n,n+1}(z) \}$, defined as

(3.8) $$\tilde{\mathbf{F}}_{n,n+1}(z) = \tilde{\Sigma}_{n,n+1}\Phi_{n,n+1}(z) = \frac{1}{\mathbf{t}_{n,n+1}} \begin{bmatrix} 1 & -\mathbf{r}_{n,n+1} \\ -\mathbf{s}_{n,n+1} & 1 \end{bmatrix} \begin{bmatrix} z^{-n_n} & 0 \\ 0 & z^{n_n} \end{bmatrix}$$

and a transmission diagram (Fig. 6), in which the transfer matrix coefficients $\{ \mathbf{r}_{n,n+1}, \mathbf{s}_{n,n+1} \}$ are *transposed* from the *actual* system (see (2.1) and Fig. 2).

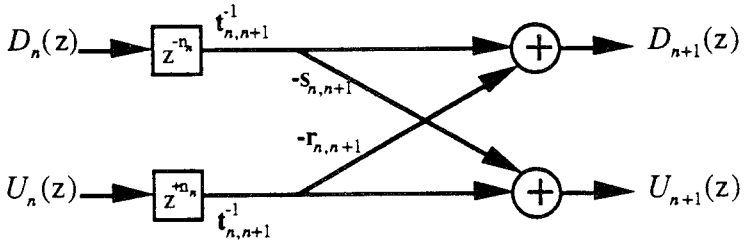


Fig. 6 Transmission diagram for discrete adjoint system

Note for the lossless system $r_{n,n+1} = s_{n,n+1}$, in which case the actual and adjoint systems are identical.

As one might suspect, there is a relationship between the scattering responses of the actual and adjoint systems. We begin by using Fig. 5 to write the layer scattering matrix $\{ S_{n,n+1}(z) \}$ for the actual system:

$$S_{n,n+1}(z) = \begin{bmatrix} \sqrt{1 - r_{n,n+1}s_{n,n+1}}\, z^{-n_n} & -s_{n,n+1} \\ +r_{n,n+1} z^{-2n_n} & \sqrt{1 - r_{n,n+1}s_{n,n+1}}\, z^{-n_n} \end{bmatrix} \quad \Rightarrow \tag{3.9a}$$

$$S^{-1}_{n,n+1}(z) = z^{+2n_n} \begin{bmatrix} \sqrt{1 - r_{n,n+1}s_{n,n+1}}\, z^{-n_n} & +s_{n,n+1} \\ -r_{n,n+1} z^{-2n_n} & \sqrt{1 - r_{n,n+1}s_{n,n+1}}\, z^{-n_n} \end{bmatrix}. \tag{3.9b}$$

Next, take the transpose and replace z with z^{-1}, giving

$$\left[S^{-1}_{n,n+1}(z^{-1}) \right]^T = \begin{bmatrix} \sqrt{1 - r_{n,n+1}s_{n,n+1}}\, z^{-n_n} & -r_{n,n+1} \\ +s_{n,n+1} z^{-2n_n} & \sqrt{1 - r_{n,n+1}s_{n,n+1}}\, z^{-n_n} \end{bmatrix}. \tag{3.10}$$

Note that (3.10) is equivalent to interchanging the transfer coefficients $\{ r_{n,n+1},\ s_{n,n+1} \}$ in (3.9a), so that $\left[S^{-1}_{n,n+1}(z^{-1}) \right]^T$ is the scattering matrix for one layer of the adjoint system, as shown in Fig. 7.

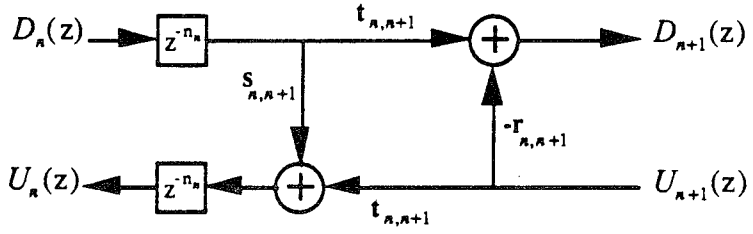


Fig. 7 Scattering diagram for adjoint system

Using a similar argument on (3.1) shows that the scattering matrix $\tilde{S}(z)$ for the entire adjoint medium is related to the scattering matrix $S(z)$ for the entire actual medium by:

$$\tilde{\mathbf{S}}(z)=\begin{bmatrix}\tilde{T}^f(z) & \tilde{R}^b(z)\\ \tilde{R}^f(z) & \tilde{T}^b(z)\end{bmatrix}=\left[\mathbf{S}^{-1}(z^{-1})\right]^{\mathrm{T}}. \tag{3.11}$$

The continuous-medium counterpart to (3.11), with Fourier frequency ω instead of z, appeared in [5]. However, (3.11) is much more useful, since the adjoint system responses can be computed exactly using a simple inverse z-transform, or the discrete Fourier transform, instead of an inverse continuous Fourier transform, which can never be computed exactly from real data.

The reciprocity relationship still holds, and we have the following identity:

$$\tilde{T}^f(z)=\tilde{T}^b(z)\Rightarrow DET\left\{\mathbf{F}(z^{-1})\right\}=F_{11}(z^{-1})F_{22}(z^{-1})-F_{12}(z^{-1})F_{21}(z^{-1})=1. \tag{3.12}$$

The transition matrix $\{\tilde{\mathbf{M}}_t(z)\}$ for the adjoint system, which acts as the z-transform of the discrete matrix Green's function of the adjoint medium, can now be found from the transition matrix $\{\mathbf{M}_t(z)\}$ elements of the actual system:

$$\tilde{\mathbf{M}}_t(z)=\prod_{i=0}^{t-1}\tilde{\mathbf{F}}_{i,i+1}(z)=\begin{bmatrix}F_{22}(z^{-1}) & F_{21}(z^{-1})\\ F_{12}(z^{-1}) & F_{11}(z^{-1})\end{bmatrix}. \tag{3.13}$$

The reflection and transmission responses of the adjoint system (defined in (3.11)) can be computed from $\{\mathbf{r}_{n,n+1},\ \mathbf{s}_{n,n+1}\}$ as in Section 3.3, using $\tilde{\mathbf{M}}_t(z)$ in place of $\mathbf{M}_t(z)$.

3.4. Data Feasibility and Sufficiency.

3.4A. Data Feasibility. We now derive another new relationship between the responses of the actual system and those of the adjoint system. We now assume that both actual and adjoint systems are both probed from just beneath free (perfectly reflecting) surfaces. This is approximately the case if there is a huge impedance mismatch at the interface of the top two layers,

such as a ground-air interface or an conductor-dielectric interface. The free-surface system equations for the actual and adjoint system are

$$\text{(3.14a)} \qquad \begin{bmatrix} T(z) \\ 0 \end{bmatrix} = \mathbf{M}_t(z) \begin{bmatrix} 1+R(z) \\ R(z) \end{bmatrix}; \qquad \text{(3.14b)} \qquad \begin{bmatrix} \tilde{T}(z) \\ 0 \end{bmatrix} = \tilde{\mathbf{M}}_t(z) \begin{bmatrix} 1+\tilde{R}(z) \\ \tilde{R}(z) \end{bmatrix},$$

where $R(z)$ and $\tilde{R}(z)$ are the free-surface impulse reflection responses of the actual and adjoint media, respectively, and $D_0(z) = U_0(z) + 1$ states that the free-surface reflects the upgoing wave to become the downgoing wave ($1 = Z\{\delta(t)\}$ is the probing impulse).

Equations (3.5), (3.12), and (3.14) can be combined to show that *the cross-correlation of the actual and adjoint medium transmission responses is the superposition of the actual and time-reversed adjoint system free-surface reflection responses*, i.e.,

$$\text{(3.15)} \qquad 1 + R(z) + \tilde{R}(z^{-1}) = T(z)\tilde{T}(z^{-1}).$$

This interesting result can be viewed as a generalization of the famous Kunetz equation from lossless to lossy systems. Kunetz [11] showed that for a lossless system, the free-surface reflection response is one side of the autocorrelation of the transmission response, i.e.,

$$\text{(3.16)} \qquad 1 + R(z) + R(z^{-1}) = T(z)T(z^{-1}).$$

The relationship between (3.15) and (3.16) can be easily seen by recalling that for a lossless system the actual and adjoint systems are identical. Note that (3.16) does not hold for lossy medium responses.

The Kunetz relation (3.16) shows that the two-sided free-surface reflection response (the probing impulse, plus the reflection response, plus its time reversal) must be a *positive definite* (**pd**) function, i.e., positive for $|z| = 1$ (z on the unit circle). This an important check on the

feasibility of the reflection data: if the data are corrupted by noise such that $1+R(z)+R(z^{-1})$ is no longer **pd**, the *data are infeasible*, i.e., there is no medium that could produce this reflection response. This is why layer stripping algorithms have the reputation of being unstable in noise; they are being fed infeasible data! If the noisy data are altered to make $1+R(z)+R(z^{-1})$ **pd**, the algorithms are stable.

We now consider the possible use of (3.15) to perform similar feasibility tests on data from *lossy* systems. Eq. (3.15) can easily be symmetrized into

$$1+\frac{1}{2}\left[R(z)+R(z^{-1})+\tilde{R}(z)+\tilde{R}(z^{-1})\right]=\frac{1}{2}\left[T(z)\tilde{T}(z^{-1})+T(z^{-1})\tilde{T}(z)\right] \tag{3.17a}$$

$$=\frac{1}{2}\left[T(z)T(z^{-1})+\tilde{T}(z)\tilde{T}(z^{-1})-\left(T(z)-\tilde{T}(z)\right)\left(T(z^{-1})-\tilde{T}(z^{-1})\right)\right] \tag{3.17b}$$

$$=\frac{1}{4}\left[\left(T(z)+\tilde{T}(z)\right)\left(T(z^{-1})+\tilde{T}(z^{-1})\right)-\left(T(z)-\tilde{T}(z)\right)\left(T(z^{-1})-\tilde{T}(z^{-1})\right)\right]. \tag{3.17c}$$

It is clear that each term on the right side is **pd**, so the symmetrized left side (compare to (3.16)) is **pd** if the difference of the **pd** terms on the right side is **pd**. This will happen if $T(z)$ and $\tilde{T}(z)$ are not too dissimilar. Specifically, if for all z on the unit circle the cosine of the phase difference between $T(z)$ and $\tilde{T}(z)$ is positive, then the left side must be **pd**. If the former condition were known to be true *a priori* (e.g., the medium has low losses), the latter condition could be used as a feasibility check on the data $R(z)$ and $\tilde{R}(z)$.

A very important issue that arises when the reflection data are noisy is distinguishing reflections off actual interfaces from noise spikes in the reflection data that could be interpreted incorrectly as interfaces. Several authors, e.g., [12], have proposed thresholding the reflection data, i.e., a data value above the threshold is interpreted as an actual reflection, while a data value below the threshold is regarded as noise and set to zero. This approach has proven quite promising for lossless media, and we propose its use for lossy media as well.

3.4B. Data Sufficiency. In the next section, we show that if the free-surface responses $\{ R(z), \tilde{R}(z) \}$ of a discrete system are known, fast reconstruction is possible using the Levinson or Schur algorithms. We now show that it is straightforward to relate the actual and adjoint *free-surface* responses $\{ R(z), \tilde{R}(z) \}$ and the measurable *half-space* impulse responses $\{ R^f(z), R^b(z), T^f(z), \text{and } T^b(z) \}$. The significance of this result is that knowledge of the four half-space actual system responses is sufficient to reconstruct the system, and anything less is insufficient. This extends a result of [5] to discrete lossy systems.

The free-surface response $\{ R(z) \}$ for the actual system can be found simply from the half-space forward reflection response $\{ R^f(z) \}$ by

$$R(z) = \frac{R^f(z)}{1 - R^f(z)}, \tag{3.18}$$

which is easily understood in terms of feedback caused by the perfectly-reflecting free surface. A relation analogous to (3.18) holds for the adjoint system as well. However, the adjoint half-space reflection response is not known. Hence, determining the free-surface reflection response $\{ \tilde{R}(z) \}$ of the adjoint system requires knowledge of all four of the actual system half-space impulse responses. Using (3.11) gives

$$\tilde{R}(z) = \frac{\tilde{R}^f(z)}{1 - \tilde{R}^f(z)} = \frac{-R^b(z^{-1})}{T^b(z^{-1})T^f(z^{-1}) + R^b(z^{-1})\left(1 - R^f(z^{-1})\right)}. \tag{3.19}$$

Therefore, even though the responses of the adjoint system are not directly measurable, we can calculate these responses from the forward and backward (i.e., *two-sided*) half-space impulse responses of the actual system.

3.5. Discrete Layer-Removal Formulae. We now show that the free-surface responses for the discrete lossy system satisfy a recursive relationship. We begin by rewriting (2.1) as

$$\frac{U_{n+1}(z)}{D_{n+1}(z)} = \frac{z^{+n_n} U_n(z) - \mathbf{r}_{n,n+1} z^{-n_n} D_n(z)}{z^{-n_n} D_n(z) - \mathbf{s}_{n,n+1} z^{+n_n} U_n(z)}, \tag{3.20}$$

and define the reflection response at the top of the n^{th} layer as $\mathrm{R}_n(z) = \dfrac{U_n(z)}{D_n(z)}$. This definition results in the following recursion for a *discrete lossy* system:

$$\mathrm{R}_{n+1}(z) = \frac{z^{+2n_n} \mathrm{R}_n(z) - \mathbf{r}_{n,n+1}}{1 - \mathbf{s}_{n,n+1} z^{+2n_n} \mathrm{R}_n(z)}. \tag{3.21}$$

The recursion is initialized at the surface using the reflection response $R_f(z)$. Each recursion removes the effect of one layer of the medium.

The lossless version of this equation has been called the *layer-removal* formula; (3.21), which is new, generalizes the formula for lossless media to lossy media. Note the feedback term in the denominator, which accounts for the multiple reflections.

A similar equation for the reflection response $\{ \tilde{\mathrm{R}}_n(z) \}$ of the adjoint system may be found by swapping the transfer matrix coefficients $\{ \mathbf{r}_{n,n+1}, \mathbf{s}_{n,n+1} \}$, yielding

$$\tilde{\mathrm{R}}_{n+1}(z) = \frac{z^{+2n_n} \tilde{\mathrm{R}}_n(z) - \mathbf{s}_{n,n+1}}{1 - \mathbf{r}_{n,n+1} z^{+2n_n} \tilde{\mathrm{R}}_n(z)}, \tag{3.22}$$

which is initialized using $\tilde{R}_f(z)$.

Since an impulse incident at the top of a discrete system requires a finite travel time to reach deeper layers, we find that the transfer coefficients $\{ \mathbf{r}_{n,n+1}, \mathbf{s}_{n,n+1} \}$ may be obtained from the reflection responses $\{ \mathrm{R}_n(z), \tilde{\mathrm{R}}_n(z) \}$, respectively, using the initial value theorem:

(3.23a) $\mathbf{r}_{n,n+1} = \lim_{z\to\infty} z^{+2n_n} \mathrm{R}_n(z)$; (3.23b) $\mathbf{s}_{n,n+1} = \lim_{z\to\infty} z^{+2n_n} \tilde{\mathrm{R}}_n(z)$.

This constitutes a simple solution to the inverse scattering problem defined next. Note that (3.21) and (3.22), run in decreasing n and initialized with zero, can also be used to solve the forward scattering problem, instead of (2.1), (3.1), (3.2), and (3.7).

4. Inverse Scattering.

4.1. Discrete Matrix Green's Function. Label the elements of the transition matrices $\{ \mathbf{M}_t(z), \tilde{\mathbf{M}}_t(z) \}$ for portions of the actual and adjoint systems using polynomials $G(z)$, $H(z)$, $J(z)$, and $K(z)$:

(4.1a) $$\mathbf{M}_t(z) = \begin{bmatrix} F_{11}(z) & F_{12}(z) \\ F_{21}(z) & F_{22}(z) \end{bmatrix} = \frac{1}{\sigma_{t-1}} \begin{bmatrix} z^{-\tau_{t-1}} G_{t-1}(z) & -z^{+\tau_{t-1}} J_{t-1}(z^{-1}) \\ -z^{-\tau_{t-1}} H_{t-1}(z) & z^{+\tau_{t-1}} K_{t-1}(z^{-1}) \end{bmatrix}, \quad t \geq 1$$

(4.1b) $$\tilde{\mathbf{M}}_t(z) = \begin{bmatrix} F_{22}(z^{-1}) & F_{21}(z^{-1}) \\ F_{12}(z^{-1}) & F_{11}(z^{-1}) \end{bmatrix} = \frac{1}{\sigma_{t-1}} \begin{bmatrix} z^{-\tau_{t-1}} K_{t-1}(z) & -z^{+\tau_{t-1}} H_{t-1}(z^{-1}) \\ -z^{-\tau_{t-1}} J_{t-1}(z) & z^{+\tau_{t-1}} G_{t-1}(z^{-1}) \end{bmatrix}, \quad t \geq 1$$

where the cumulative delay $\{ \tau_{t-1} \}$ and transmission factor $\{ \sigma_t \}$ are defined by

(4.2a) $\tau_{t-1} = \sum_{i=0}^{t-1} \mathrm{n}_i$; (4.2b) $\sigma_{t-1} = \prod_{i=0}^{t-1} \mathbf{t}_{i,i+1} = \prod_{i=0}^{t-1} \sqrt{1 - \mathbf{r}_{i,i+1}\mathbf{s}_{i,i+1}}$.

The matrices $\mathbf{M}_n(z)$ and $\tilde{\mathbf{M}}_n(z)$ are the discrete matrix Green's functions for the portion of the medium above (or to left of) the n^{th} interface. Note that $\mathbf{M}_n(z) = \mathbf{F}_{n,n+1}(z)\mathbf{M}_{n-1}(z)$ and $\tilde{\mathbf{M}}_n(z) = \tilde{\mathbf{F}}_{n,n+1}(z)\tilde{\mathbf{M}}_{n-1}(z)$; therefore,

(4.3a) $$\begin{bmatrix} z^{-\tau_n} G_n(z) - z^{+\tau_n} J_n(z^{-1}) \\ z^{-\tau_n} H_n(z) - z^{+\tau_n} K_n(z^{-1}) \end{bmatrix} = \begin{bmatrix} 1 & \mathbf{s}_{n,n+1} \\ \mathbf{r}_{n,n+1} & 1 \end{bmatrix} \begin{bmatrix} z^{-n_n} & 0 \\ 0 & z^{+n_n} \end{bmatrix} \begin{bmatrix} z^{-\tau_{n-1}} G_{n-1}(z) - z^{+\tau_{n-1}} J_{n-1}(z^{-1}) \\ z^{-\tau_{n-1}} H_{n-1}(z) - z^{+\tau_{n-1}} K_{n-1}(z^{-1}) \end{bmatrix}$$

$$(4.3b)\quad \begin{bmatrix} z^{-\tau_n}K_n(z)-z^{+\tau_n}H_n(z^{-1}) \\ z^{-\tau_n}J_n(z)-z^{+\tau_n}G_n(z^{-1}) \end{bmatrix} = \begin{bmatrix} 1 & r_{n,n+1} \\ s_{n,n+1} & 1 \end{bmatrix} \begin{bmatrix} z^{-n_n} & 0 \\ 0 & z^{+n_n} \end{bmatrix} \begin{bmatrix} z^{-\tau_{n-1}}K_{n-1}(z)-z^{+\tau_{n-1}}H_{n-1}(z^{-1}) \\ z^{-\tau_{n-1}}J_{n-1}(z)-z^{+\tau_{n-1}}G_{n-1}(z^{-1}) \end{bmatrix}.$$

Equations (4.3a) and (4.3b) are initialized by noting that for *no* interfaces $\{ t=0,\ \tau_{t-1}=0 \}$ we have $\mathbf{M}_{-1}(z)=\tilde{\mathbf{M}}_{-1}(z)=\mathbf{I}$, so that $G_{-1}(z)=K_{-1}(z)=1$ and $H_{-1}(z)=J_{-1}(z)=0$. Note for a *lossless* system $G(z)=K(z)$, $H(z)=J(z)$, and $\mathbf{M}_t(z)=\tilde{\mathbf{M}}_t(z)$.

4.2. Derivation of Asymmetric Toeplitz System. We are again interested in the waves generated in the actual and adjoint systems due to a downgoing impulse in a medium having a free-surface upper boundary. Wave components at the interface of the $(n-1)^{th}$ and n^{th} layers in the actual system $\{ D_n^R(z),\ U_n^R(z) \}$ and adjoint system $\{ \tilde{D}_n^R(z),\ \tilde{U}_n^R(z) \}$ are related to the actual $\{ R(z) \}$ and adjoint $\{ \tilde{R}(z) \}$ free-surface reflection responses using the discrete matrix Green's functions (4.1):

$$(4.4a)\quad \begin{bmatrix} D_n^R(z) \\ U_n^R(z) \end{bmatrix} = \frac{1}{\sigma_{n-1}} \begin{bmatrix} z^{-\tau_{n-1}}G_{n-1}(z) & -z^{+\tau_{n-1}}J_{n-1}(z^{-1}) \\ -z^{-\tau_{n-1}}H_{n-1}(z) & z^{+\tau_{n-1}}K_{n-1}(z^{-1}) \end{bmatrix} \begin{bmatrix} 1+R(z) \\ R(z) \end{bmatrix};$$

$$(4.4b)\quad \begin{bmatrix} \tilde{D}_n^R(z) \\ \tilde{U}_n^R(z) \end{bmatrix} = \frac{1}{\sigma_{n-1}} \begin{bmatrix} z^{-\tau_{n-1}}K_{n-1}(z) & -z^{+\tau_{n-1}}H_{n-1}(z^{-1}) \\ -z^{-\tau_{n-1}}J_{n-1}(z) & z^{+\tau_{n-1}}G_{n-1}(z^{-1}) \end{bmatrix} \begin{bmatrix} 1+\tilde{R}(z) \\ \tilde{R}(z) \end{bmatrix}.$$

We solve for $R(z)$ in the top equation of (4.4a) and $R(z^{-1})$ in the bottom. Similarly, we solve for $\tilde{R}(z)$ in the top equation of (4.4b) and $\tilde{R}(z^{-1})$ in the bottom. This yields

$$(4.5a)\quad R(z)=\frac{\sigma_{n-1}D_n^R(z)-z^{-\tau_{n-1}}G_{n-1}(z)}{z^{-\tau_{n-1}}G_{n-1}(z)-z^{+\tau_{n-1}}J_{n-1}(z^{-1})};\qquad (4.5b)\quad R(z^{-1})=\frac{\sigma_{n-1}U_n^R(z^{-1})+z^{+\tau_{n-1}}H_{n-1}(z^{-1})}{z^{-\tau_{n-1}}K_{n-1}(z)-z^{+\tau_{n-1}}H_{n-1}(z^{-1})};$$

$$(4.5c)\quad \tilde{R}(z)=\frac{\sigma_{n-1}\tilde{D}_n^R(z)-z^{-\tau_{n-1}}K_{n-1}(z)}{z^{-\tau_{n-1}}K_{n-1}(z)-z^{+\tau_{n-1}}H_{n-1}(z^{-1})};\qquad (4.5d)\quad \tilde{R}(z^{-1})=\frac{\sigma_{n-1}\tilde{U}_n^R(z^{-1})+z^{+\tau_{n-1}}J_{n-1}(z^{-1})}{z^{-\tau_{n-1}}G_{n-1}(z)-z^{+\tau_{n-1}}J_{n-1}(z^{-1})}.$$

Adding (4.5a) to (4.5d) and (4.5b) to (4.5c) (replacing $z \rightarrow z^{-1}$ in the latter) yields:

$$\left(1+R(z)+\tilde{R}(z^{-1})\right)\left(z^{-\tau_{n-1}}G_{n-1}(z)-z^{+\tau_{n-1}}J_{n-1}(z^{-1})\right)=\sigma_{n-1}\left(D_n^R(z)+\tilde{U}_n^R(z^{-1})\right); \tag{4.6a}$$

$$\left(1+R(z)+\tilde{R}(z^{-1})\right)\left(z^{+\tau_{n-1}}K_{n-1}(z^{-1})-z^{-\tau_{n-1}}H_{n-1}(z)\right)=\sigma_{n-1}\left(\tilde{D}_n^R(z^{-1})+U_n^R(z)\right). \tag{4.6b}$$

Note that causality holds for both the actual and adjoint systems; therefore the waves $D_n^R(z)$, $U_n^R(z)$, $\tilde{D}_n^R(z)$, and $\tilde{U}_n^R(z)$ are all zero in the time domain until the probing impulse arrives. At the moment of arrival $\{\tau_{n-1}\}$ of the probing impulse, the downgoing waves $\{D_n^R(z),\ \tilde{D}_n^R(z)\}$ are both σ_{n-1}, and the upgoing waves $\{U_n^R(z),\ \tilde{U}_n^R(z)\}$ are both zero; n_n seconds later the upgoing waves are $\sigma_{n-1}\mathbf{r}_{n-1,n}$ and $\sigma_{n-1}\mathbf{s}_{n-1,n}$, respectively. Using these facts, and equating powers of z in (4.6), we can write (4.6) in the time-domain as

$$\begin{bmatrix} 1 & r(1) & r(2) & \cdot & \cdot & r(n) \\ \tilde{r}(1) & 1 & r(1) & & & \cdot \\ \tilde{r}(2) & \tilde{r}(1) & 1 & \cdot & & \cdot \\ \cdot & & \cdot & \cdot & \cdot & \cdot \\ \cdot & & & \cdot & \cdot & r(1) \\ \tilde{r}(n) & \cdot & \cdot & \cdot & \tilde{r}(1) & 1 \end{bmatrix} \begin{bmatrix} g_n(0)-j_n(n) & k_n(n)-h_n(0) \\ g_n(1)-j_n(n-1) & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & k_n(1)-h_n(n-1) \\ g_n(n)-j_n(0) & k_n(0)-h_n(n) \end{bmatrix} = \begin{bmatrix} \sigma_n & 0 \\ 0 & \cdot \\ \cdot & \cdot \\ & \\ \cdot & 0 \\ 0 & \sigma_n \end{bmatrix}, \tag{4.7a}$$

$$\mathbf{r}_{n-1,n}=-\sum_{i=0}^{n}\frac{\tilde{r}(i+1)(k_n(n-i)-h_n(i))}{\sigma_{n-1}}, \tag{4.7b}$$

$$\mathbf{s}_{n-1,n}=-\sum_{i=0}^{n}\frac{r(n+1-i)(g_n(i)-j_n(n-i))}{\sigma_{n-1}}, \tag{4.7c}$$

where $n=\tau_n$ (recall we are letting $\Delta=1$ without loss of generality), $G_n(z)=\sum_{i=0}^{n} g_n(i)\,z^i$ and similarly for $H_n(z)$, $J_n(z)$, and $K_n(z)$. By examining the propagation of coefficients z^0 and $z^{-\tau_n}$ in (4.4), it can be shown that $g_n(0)=k_n(0)=1$ and $h_n(n)=j_n(n)=0$. Although it might seem

that many entries in (4.7a) are zero, since reflections can only reach the surface at integer linear combinations of the τ_i, in point of fact most entries are non-zero due to multiple reflections.

Eq. (4.7) shows the equivalence of the computation of the elements of the transition matrices $\mathbf{M}_n(z)$ and $\tilde{\mathbf{M}}_n(z)$ (the z-transform of the discrete matrix Green's function) and the solution of two asymmetric Toeplitz systems which act as discrete counterparts to integral equations. In (4.7) the discrete-time free-surface reflection responses $\{r(n),\ \tilde{r}(n)\}$ from the actual and adjoint systems are assembled into an asymmetric Toeplitz matrix, which is solved to recover the transfer coefficients $\{\mathbf{r}_{n-1,n},\ \mathbf{s}_{n-1,n}\}$ at each interface. A solution by Gaussian elimination of the $n \times n$ system of equations $\mathbf{A}_{n\times n}\mathbf{x}_n = \mathbf{b}_n$ requires $\mathbf{O}(n^3)$ multiplications and additions.

4.3. Solution using Asymmetric Levinson and Schur Algorithms. An $n \times n$ Toeplitz system of equations can be solved recursively using the Levinson or Schur algorithms, which requires $\mathbf{O}(n^2)$ multiplications and additions. For ease of notation and familiarity, we make the following substitutions:

(4.8a) $$r_n \rightarrow r(n)$$

(4.8b) $$k_n^1 \rightarrow \mathbf{s}_{n-1,n} \qquad k_n^2 \rightarrow \mathbf{r}_{n-1,n}$$

(4.8c) $$t_n = t_{n-1}(1-k_n^1 k_n^2) \rightarrow \sigma_n = \sigma_{n-1}\sqrt{1-\mathbf{r}_{n-1,n}\mathbf{s}_{n-1,n}}$$

(4.8d) $$A_n(z) \rightarrow z^{-\tau_0}G_n(z) - z^{+\tau_0}J_n(z^{-1}) \Rightarrow a_{n,i} \rightarrow g_n(i) - j_n(n-i)$$

(4.8e) $$B_n(z) \rightarrow z^{+\tau_0}K_n(z^{-1}) - z^{-\tau_0}H_n(z) \Rightarrow b_{n,i} \rightarrow k_n(n-i) - h_n(i)$$

<u>*Asymmetric Levinson algorithm*</u> [13]

Step 1. *Initialize:* $n = 0,\ A_0(z) = B_0(z) = 1,$ *and* $t_0 = r_0.$

Step 2. $$k_{n+1}^1 = -\sum_{i=0}^{n} \frac{a_{n,i}r_{n+1-i}}{t_n} \text{ and } k_{n+1}^2 = -\sum_{i=0}^{n} \frac{b_{n,i}r_{-(i+1)}}{t_n}$$

Step 3. $$\begin{bmatrix} A_{n+1}(z) \\ B_{n+1}(z) \end{bmatrix} = \begin{bmatrix} 1 & k^1_{n,n+1} z \\ k^2_{n,n+1} & z \end{bmatrix} \begin{bmatrix} A_n(z) \\ B_n(z) \end{bmatrix}$$

Step 4. $$t_{n+1} = t_n \left(1 - k^1_{n+1} k^2_{n+1}\right)$$

Step 5. *Increment* n, *return to Step 2.*

Note that the main recursion (Step 3) is identical to (4.3a), after the substitutions (4.8). In fact, the Levinson algorithm, applied to this problem, is simply a recursive implementation of (4.3a). For the lossless case, we note that (4.7) is symmetric, and thus the symmetric Levinson algorithm may be used.

If we are not interested in recovering the components of the transition matrices, but only interested in the transfer coefficients $\{ \mathbf{r}_{n,n+1}, \mathbf{s}_{n,n+1} \}$, we may use the asymmetric Schur algorithm:

Asymmetric Schur algorithm [13]

Step 1. *Initialize:* $n = 0$, $D_0(z) = U_0(z) = \mathrm{R}(z)$,
$\tilde{D}_0(z) = \tilde{U}_0(z) = \tilde{\mathrm{R}}(z)$.

Step 2. $$\begin{bmatrix} U_{n+1}(z) \\ D_{n+1}(z) \end{bmatrix} = \begin{bmatrix} 1 & k^1_{n,n+1} z \\ k^2_{n,n+1} & z \end{bmatrix} \begin{bmatrix} U_n(z) \\ D_n(z) \end{bmatrix}$$
$$\begin{bmatrix} \tilde{U}_{n+1}(z) \\ \tilde{D}_{n+1}(z) \end{bmatrix} = \begin{bmatrix} 1 & k^2_{n,n+1} z \\ k^1_{n,n+1} & z \end{bmatrix} \begin{bmatrix} \tilde{U}_n(z) \\ \tilde{D}_n(z) \end{bmatrix}$$

Step 3. $$k^1_{n+1} = -\frac{u_{n,n+1}}{d_{n,n}} = -\frac{\tilde{d}_{n,-1}}{\tilde{u}_{n,0}} \qquad k^2_{n+1} = -\frac{d_{n,-1}}{u_{n,0}} = \frac{\tilde{u}_{n,n+1}}{\tilde{d}_{n,n}}$$

Step 4. *Increment* n, *return to Step 2.*

As an aside, we note that our term "*adjoint system*" first appeared in [13], where it was used entirely as a linear algebraic convenience.

5. Discrete Lossy Transmission Line.

The *lossless* transmission line has been used to motivate discussion of discrete inverse problems [1]. Inverse scattering for a discrete *lossy* transmission line is a natural extension of this previous work. However, this problem is of considerable interest in its own right, since it can be used to model interchip communications [9].

5.1. Basic Equations. A segment of a discrete lossy transmission line is shown in Fig. 8. Over a finite interval $\{\Delta_n\}$, the distributed series resistance R_n, series inductance L_n, shunt conductance G_n, and shunt capacitance C_n, all per unit length, are constant. We omit the derivation of the telegrapher's equations and wave equations, since these may be found in [5] and [9].

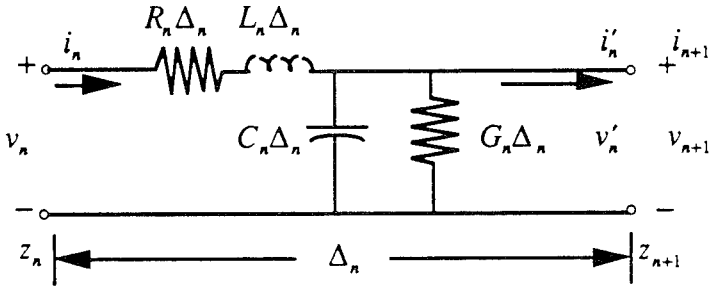


Fig. 8 Segment of a Discrete Lossy Transmission Line

The solutions $\{v,\ i\}$ to the transmission line wave equations anywhere on the line is a combination of incident $\{v^i,\ i^i\}$ and reflected $\{v^r,\ i^r\}$ waves [1]. Consider the propagation of the *energy-normalized* sinusoidal voltage wave components $\{V^i = Z^{-1/2}v^i,\ V^r = Z^{-1/2}v^r\}$ having frequency ω, where the characteristic impedance $\{Z\}$ in a segment of the lossy line is the

complex quantity $Z = \sqrt{\frac{R + j\omega L}{G + j\omega C}}$. The solution of the wave equations in ter[illegible] of the energy-normalized wave components $\{V^i,\ V^r\}$ at a point z on the line is

(5.1a) $v = Z^{\frac{1}{2}}\left(V^i e^{j\omega x} e^{\frac{-j\omega x}{v}} + V^r e^{j\omega x} e^{\frac{j\omega x}{v}} \right)$; (5.1b) $i = Z^{-\frac{1}{2}}\left(V^i e^{j\omega x} e^{\frac{-j\omega x}{v}} - V^r e^{j\omega x} e^{\frac{j\omega x}{v}} \right)$,

where the *wave velocity* $v = \left(\left(\frac{R}{j\omega} + L \right)\left(\frac{G}{j\omega} + C \right) \right)^{-\frac{1}{2}}$ (note the complex quantity v includes propagation, attenuation, and dispersion effects). Our goal is to find the relationship between wave components $\{V_n^i,\ V_n^r\}$ of one section and those $\{V_{n+1}^i,\ V_{n+1}^r\}$ of the next.

5.2. Formulation.

5.2A. Interface Effect. At an interface at position $z = z_{n+1} = z_n + \Delta_n$ between two sections having dissimilar impedances $Z_n \neq Z_{n+1}$, the voltages and currents must be continuous by Kirchoff's laws, i.e., $v_n' = v_{n+1}$ and $i_n' = i_{n+1}$. Using (5.1) and equating voltages and currents on the opposite sides of the interface, we have that the wave components $\{V_{n+1}^i,\ V_{n+1}^r\}$ just to the right of the interface at $z = z_{n+1}$ are related to the components $\{V_n'^i,\ V_n'^r\}$ just to the left of the interface at $z = z_{n+1}$ by

(5.2)
$$\begin{bmatrix} V_{n+1}^i e^{\frac{-j\omega z_{n+1}}{v_{n+1}}} \\ V_{n+1}^r e^{\frac{j\omega z_{n+1}}{v_{n+1}}} \end{bmatrix} = \frac{1}{\sqrt{1-\Gamma_{n,n+1}^2}} \begin{bmatrix} 1 & -\Gamma_{n,n+1} \\ -\Gamma_{n,n+1} & 1 \end{bmatrix} \begin{bmatrix} V_n'^i e^{\frac{-j\omega z_{n+1}}{v_n}} \\ V_n'^r e^{\frac{j\omega z_{n+1}}{v_n}} \end{bmatrix},$$

where $\Gamma_{n,n+1} = \frac{Z_{n+1} - Z_n}{Z_{n+1} + Z_n}$ is the reflection coefficient between the n^{th} and $(n+1)^{st}$ layers, i.e., for the interface at $z = z_{n+1}$.

5.2B. Homogeneous Segment Effect. Since $\Gamma \neq 0$ only at an interface, within the homogeneous transmission line segment $\{ z \in (z_n, z_{n+1}) \}$, the wave components $\{ V_n'^i,\ V_n'^r \}$ at the end of the section are related to those $\{ V_n^i,\ V_n^r \}$ at the beginning by the *propagation delay* through the layer. The propagation constant $\{ \frac{j\omega}{v_n} \}$ can be expressed as the complex sum of phase $\{ \beta_n \}$ and attenuation $\{ \alpha_n \}$ constants, i.e., $\frac{j\omega}{v_n} = j\beta_n + \alpha_n$. In matrix form, we have

$$(5.3) \quad \begin{bmatrix} V_n'^i e^{\frac{-j\omega z_{n+1}}{v_0}} \\ V_n'^r e^{\frac{j\omega z_{n+1}}{v_0}} \end{bmatrix} = \begin{bmatrix} e^{\frac{-j\omega\Delta_n}{v_0}} & 0 \\ 0 & e^{\frac{j\omega\Delta_n}{v_0}} \end{bmatrix} \begin{bmatrix} V_n^i e^{\frac{-j\omega z_n}{v_0}} \\ V_n^r e^{\frac{j\omega z_n}{v_0}} \end{bmatrix} = \begin{bmatrix} e^{-\alpha_n\Delta_n} & 0 \\ 0 & e^{+\alpha_n\Delta_n} \end{bmatrix} \begin{bmatrix} e^{-j\beta_n\Delta_n} & 0 \\ 0 & e^{+j\beta_n\Delta_n} \end{bmatrix} \begin{bmatrix} V_n^i e^{\frac{-j\omega z_n}{v_0}} \\ V_n^r e^{\frac{j\omega z_n}{v_0}} \end{bmatrix}.$$

5.2C. Combined Effect. Combing (5.2) and (5.3) gives the relationship between the wave components of one layer $\{ V_n^i,\ V_n^r \}$ and those in the next $\{ V_{n+1}^i,\ V_{n+1}^r \}$:

$$(5.4) \quad \begin{bmatrix} V_{n+1}^i e^{\frac{-j\omega z_{n+1}}{v_{0+1}}} \\ V_{n+1}^r e^{\frac{j\omega z_{n+1}}{v_{0+1}}} \end{bmatrix} = \frac{1}{\sqrt{1-\Gamma_{n,n+1}^2}} \begin{bmatrix} e^{-\alpha_n\Delta_n} & -\Gamma_{n,n+1}e^{\alpha_n\Delta_n} \\ -\Gamma_{n,n+1}e^{-\alpha_n\Delta_n} & e^{\alpha_n\Delta_n} \end{bmatrix} \begin{bmatrix} e^{-j\beta_n\Delta_n} & 0 \\ 0 & e^{j\beta_n\Delta_n} \end{bmatrix} \begin{bmatrix} V_n^i e^{\frac{-j\omega z_n}{v_0}} \\ V_n^r e^{\frac{j\omega z_n}{v_0}} \end{bmatrix}.$$

Since it has zero length, the interface itself is lossless; we therefore have $e^{-\alpha_{n+1}z_{n+1}} = e^{-\alpha_n z_{n+1}} = e^{-\alpha_n z_n} e^{-\alpha_n\Delta_n}$. Using this relationship, we introduce the phasor wave components $\{ \mathbf{V}_n^i,\ \mathbf{V}_n^r \}$ as

$$(5.5a) \quad V_{n+1}^i e^{\frac{-j\omega z_{n+1}}{v_{n+1}}} = V_{n+1}^i e^{-j\beta_{n+1}z_{n+1}} e^{-\alpha_{n+1}z_{n+1}} = V_{n+1}^i e^{-j\beta_{n+1}z_{n+1}} e^{-\alpha_n z_n} e^{-\alpha_n\Delta_n} = \mathbf{V}_{n+1}^i e^{-\prod_{i=0}^{n}\alpha_i\Delta_i};$$

$$(5.5b) \quad V_{n+1}^r e^{\frac{+j\omega z_{n+1}}{v_{n+1}}} = V_{n+1}^r e^{+j\beta_{n+1}z_{n+1}} e^{+\alpha_{n+1}z_{n+1}} = V_{n+1}^r e^{+j\beta_{n+1}z_{n+1}} e^{+\alpha_n z_n} e^{+\alpha_n\Delta_n} = \mathbf{V}_{n+1}^r e^{+\prod_{i=0}^{n}\alpha_i\Delta_i},$$

where $e^{-\prod_{i=0}^{n}\alpha_i\Delta_i}$ is the cumulative loss of the previous layers. Substituting (5.5) into (5.4), we have our desired relationship:

$$(5.6)\qquad \begin{bmatrix} \mathbf{V}^i_{n+1} \\ \mathbf{V}^r_{n+1} \end{bmatrix} = \frac{1}{\sqrt{1-\Gamma^2_{n,n+1}}} \begin{bmatrix} 1 & -\Gamma_{n,n+1} e^{2\sum_{m=0}^{n} \alpha_m \Delta_m} \\ -\Gamma_{n,n+1} e^{-2\sum_{m=0}^{n} \alpha_m \Delta_m} & 1 \end{bmatrix} \begin{bmatrix} e^{-j\beta_n \Delta_n} & 0 \\ 0 & e^{j\beta_n \Delta_n} \end{bmatrix} \begin{bmatrix} \mathbf{V}^i_n \\ \mathbf{V}^r_n \end{bmatrix}.$$

We thus have defined the layer-by-layer wave component relationships by means of a frequency-dependent *layer propagation* matrix $\{ \mathbf{F}_{n,n+1}(\omega) = \Sigma_{n,n+1}(\omega)\Phi_{n,n+1}(\omega) \}$

$$(5.7a)\qquad \begin{bmatrix} \mathbf{V}^i_{n+1} \\ \mathbf{V}^r_{n+1} \end{bmatrix} = \mathbf{F}_{n,n+1}(\omega) \begin{bmatrix} \mathbf{V}^i_n \\ \mathbf{V}^r_n \end{bmatrix};$$

having transfer matrix

$$(5.7b)\qquad \Sigma_{n,n+1}(\omega) = \frac{1}{\sqrt{1-\Gamma^2_{n,n+1}}} \begin{bmatrix} 1 & -\Gamma_{n,n+1} e^{2\sum_{m=0}^{n} \alpha_m \Delta_m} \\ -\Gamma_{n,n+1} e^{-2\sum_{m=0}^{n} \alpha_m \Delta_m} & 1 \end{bmatrix};$$

and a *time-delay* matrix

$$(5.7c)\qquad \Phi_{n,n+1}(\omega) = \begin{bmatrix} e^{-j\beta_n \Delta_n} & 0 \\ 0 & e^{j\beta_n \Delta_n} \end{bmatrix} \Rightarrow \Phi_{n,n+1}(z) = \begin{bmatrix} z^{-n_n} & 0 \\ 0 & z^{+n_n} \end{bmatrix},$$

since the time delay through each layer is assumed to be an integer multiple of a small time increment Δ, i.e., $j\beta_n\Delta_n = j\omega\sqrt{L_nC_n}\Delta_n = j\omega n_n\Delta_n$. As a result, we can replace the Fourier kernel $e^{j\omega\Delta}$ with the z-transform kernel $z = e^{j\omega\Delta}$, since all events occur at integer multiples of Δ.

5.2D. Discrete Time Formulation. We now make either of the following two assumptions:

(1) The transmission line is *dispersionless*, meaning that the *Heaviside condition* $R_nC_n = G_nL_n$ holds, i.e., the two time constants in Fig. 8 are equal; *or*

(2) The transmission line is a *high-frequency* line, so that $j\omega L_n >> R_n$ and $j\omega C_n >> G_n$. We discuss this assumption in more detail in Section 5.3.

Either assumption implies that the impedance $\{ Z_n \}$, phase constant $\{ \beta_n \}$, and attenuation constant $\{ \alpha_n \}$, may be approximated by

$$(5.8a)\quad Z_n \approx \sqrt{\frac{L_n}{C_n}}; \qquad (5.8b)\quad \beta_n \approx \omega\sqrt{L_n C_n}; \qquad (5.8c)\quad \alpha_n \approx \frac{1}{2}\left(G_n Z_n + \frac{R_n}{Z_n} \right) or\ \sqrt{R_n G_n}.$$

We now define the *transfer* coefficients $\{ \mathbf{r}_{n,n+1},\ \mathbf{s}_{n,n+1} \}$ using the frequency-independent attenuation coefficient $\{ \alpha_n \}$ in (5.8c) as

$$(5.9a)\qquad \mathbf{r}_{n,n+1} = \Gamma_{n,n+1} e^{-2\sum_{m=0}^{n} \alpha_m \Delta_m}; \qquad (5.9b)\qquad \mathbf{s}_{n,n+1} = \Gamma_{n,n+1} e^{+2\sum_{m=0}^{n} \alpha_m \Delta_m}.$$

Note for a lossless section of transmission line $\alpha_n = 0$, so that $\mathbf{r}_{n,n+1} = \mathbf{s}_{n,n+1}$. Also note that the transmission loss is $\mathbf{t}_{n,n+1} = \sqrt{1 - \mathbf{r}_{n,n+1}\mathbf{s}_{n,n+1}} = \sqrt{1 - \Gamma^2_{n,n+1}} < 1$.

Since the time-delay matrix $\Phi_{n,n+1}(z)$ can be expressed using the z-transform, and the transfer matrix $\Sigma_{n,n+1}$ is independent of ω or z, we can combine them into the *discrete-time* layer propagation matrix

$$(5.10)\qquad \mathbf{F}_{n,n+1}(z) = \frac{1}{\sqrt{1 - \mathbf{r}_{n,n+1}\mathbf{s}_{n,n+1}}} \begin{bmatrix} 1 & -\mathbf{s}_{n,n+1} \\ -\mathbf{r}_{n,n+1} & 1 \end{bmatrix} \begin{bmatrix} z^{-n_n} & 0 \\ 0 & z^{n_n} \end{bmatrix},$$

where $\mathbf{r}_{n,n+1}$ and $\mathbf{s}_{n,n+1}$ are defined in (5.9).

Eq. (5.10) is the discrete-time counterpart of the continuous-time asymmetric two-component wave system (122) of [5]. To see this, assume the line is dispersionless and let the time increment $\Delta \to 0$, along with the layer thicknesses. The reflection coefficient $\Gamma_{n,n+1}$ becomes $m(z) = \frac{1}{2Z(z)} \frac{dZ(z)}{dz}$ ((117) of [5]) and $\mathbf{r}_{n,n+1}$ and $\mathbf{s}_{n,n+1}$ become (123) of [5]. The latter also appeared in [14], where they were reconstructed by solving a coupled pair of integral equations which are the continuous-time counterparts of the two asymmetric Toeplitz systems (4.7a).

These continuous-time equations include dispersion effects, since they are derived directly from the telegrapher's equations. However, the discretized versions of these continuous-time equations, which in effect reconstruct a discrete layered medium such as considered in the present paper, do not correctly model dispersion. Hence some analysis of the relative importance of dispersion vs. absorption is required for numerical implementation of any of these equations.

5.3. Relative Effects of Dispersion and Absorption. The assumptions of (5.8) made in Section 5.2D are tantamount to neglecting dispersion (frequency-dependent effects that alter the shape of the probing pulse) as compared to absorption (frequency-independent effects that attenuate the amplitude of, but do not alter the shape of, the probing pulse). We now show why this is a reasonable assumption.

We consider the effect of propagation within a lossy homogeneous layer. From (5.1), this is $e^{\pm j\omega\frac{z}{v}}$, where z is distance, the sign depends on upward versus downward propagation, and $v=\left[\left(L+\frac{R}{j\omega}\right)\left(C+\frac{G}{j\omega}\right)\right]^{-\frac{1}{2}}$ is the complex wave speed. Using the binomial expansion $(1+\varepsilon)^{+\frac{1}{2}}=1+\frac{\varepsilon}{2}-\frac{\varepsilon^2}{8}+O(\varepsilon^3)$, we have

$$e^{j\omega\frac{z}{v}}=\exp\left[j\omega z\sqrt{LC}\left(1+\frac{1}{j\omega}\left(\frac{R}{L}+\frac{G}{C}\right)+\frac{1}{(j\omega)^2}\frac{RG}{LC}\right)^{\frac{1}{2}}\right] \tag{5.11a}$$

$$=\exp\left[j\omega z\sqrt{LC}\left(1+\frac{1}{2j\omega}\left(\frac{R}{L}+\frac{G}{C}\right)+\frac{1}{2(j\omega)^2}\frac{RG}{LC}-\frac{1}{8(j\omega)^2}\left(\frac{R}{L}+\frac{G}{C}\right)^2+O\left(\frac{1}{\omega^3}\right)\right)\right] \tag{5.11b}$$

$$=\exp\left[j\omega z\sqrt{LC}\left(1+\frac{1}{2j\omega}\left(\frac{R}{L}+\frac{G}{C}\right)-\frac{1}{8(j\omega)^2}\left(\frac{R}{L}-\frac{G}{C}\right)^2+O\left(\frac{1}{\omega^3}\right)\right)\right] \tag{5.11c}$$

$$=\exp\left[j\omega z\sqrt{LC}\left(1+\frac{a}{j\omega}-\frac{b^2}{2(j\omega)^2}+O\left(\frac{1}{\omega^3}\right)\right)\right], \tag{5.11d}$$

where $a = \frac{1}{2}\left(\frac{R}{L} + \frac{G}{C}\right)$ and $b = \frac{1}{2}\left(\frac{R}{L} - \frac{G}{C}\right)$.

Analysis of the final bracketed expansion shows the relative significance of various propagation effects. The 0^{th}-order term $\{1\}$ multiplying $j\omega z\sqrt{LC}$ is an eikonal term, which simply expresses the fact that the wave propagates at speed $\frac{1}{\sqrt{LC}}$. The first-order term $\{\frac{a}{j\omega}\}$, after multiplication by $j\omega z\sqrt{LC}$, simply represents the absorption (note $a = \alpha\sqrt{LC}$ for α defined in (5.8)). The second-order term $\{\frac{-b^2}{2(j\omega)^2}\}$ represent the most significant effects of dispersion (note the condition for a dispersionless line is precisely $b = 0$).

The significance of this expansion is that dispersion is a higher-order asymptotic (in low ω) effect than absorption. Comparing this expansion to low-frequency asymptotic expansions in optics, we see that for high-frequency pulses, dispersion will be negligible compared to absorption and interface reflection. This is not apparent from the continuous-time development in [5], so that the discrete formulation again provides new insights.

A similar analysis can be carried out for attenuation and dispersion caused by reflection and transmission of waves at interfaces. We have

$$Z = \sqrt{\frac{R + j\omega L}{G + j\omega C}} = \sqrt{\frac{L}{C}}\left(1 + \frac{1}{2j\omega}\left(\frac{R}{L} - \frac{G}{C}\right) + O\left(\frac{1}{\omega^2}\right)\right) = \sqrt{\frac{L}{C}}\left(1 + \frac{b}{j\omega} + O\left(\frac{1}{\omega^2}\right)\right). \tag{5.12}$$

This shows that dispersion will again be a low-frequency asymptotic correction to the frequency-independent attenuation caused by $Z = \sqrt{\frac{L}{C}}$ in $\Gamma_{n,n+1} = \frac{Z_{n+1} - Z_n}{Z_{n+1} + Z_n}$. Note that for a dispersionless line ($b = 0$) this correction is again zero.

Note that discretized versions of the continuous-time equations in [5] and [6] include dispersion as a continuous scattering phenomenon, modeled by the reflectivity function $b(z)$ in [5]. Medium inhomogeneities are modeled by the reflectivity function $m(z)$ defined above; $b(z)$

and $m(z)$ are weighted equally. While this is correct in continuous time, the above analysis shows that for discretized equations, corresponding to a discretized medium, these effects should *not* be treated as having equal importance. In short, discretized continuous equations give the wrong answer; an explicitly discrete formulation is called for.

6. Electromagnetic Waves in Layered Dielectrics. We now extend the transmission line formulation to the case of electromagnetic plane waves propagating in a lossy layered dielectric. Instead of *segments* of transmission line with uniform *impedances* $\{ Z_n \}$, we now have *layers* of dielectrics having infinite lateral extent and uniform *dielectric constants* $\{ \varepsilon_n \}$. In reading this section, parallels to Section 5 should be noted throughout. The forward and inverse scattering problems here have applications in radioglaciology [7] and radar reflections off of stratified dielectrics [8]. We have applied these results to radioglaciology in [15].

6.1. Basic Equations. The general form of a *time-harmonic* propagating electromagnetic wave consists of orthogonal electric $\{ \mathbf{E} = Ee^{-j\mathbf{k}\cdot\mathbf{r}} \}$ and magnetic $\{ \mathbf{H} = He^{-j\mathbf{k}\cdot\mathbf{r}} \}$ field vectors having amplitudes $\{ E = \mathrm{E}e^{j\omega t} \}$ and $\{ H = \mathrm{H}e^{j\omega t} \}$, respectively, and frequency $\{ \omega \}$. The propagating waves $\{ \mathbf{E}, \mathbf{H} \}$ each have a temporal $\{ \omega t \}$ and spatial $\{ \mathbf{k}\cdot\mathbf{r} \}$ component, where $\mathbf{k} = k_x\hat{\mathbf{x}} + k_y\hat{\mathbf{y}} + k_z\hat{\mathbf{z}}$ is the propagation vector and $\mathbf{r} = x\hat{\mathbf{x}} + y\hat{\mathbf{y}} + z\hat{\mathbf{z}}$ is the displacement vector. The magnitude of the propagation vector is the *wavenumber* or *propagation constant* and is given by $k = |\mathbf{k}| = \omega\sqrt{\mu\varepsilon}$. By the consistency condition, $k = \sqrt{k_x^2 + k_y^2 + k_z^2}$.

In general, the dielectric constant $\{ \varepsilon \}$ is complex; it may be expressed as $\varepsilon = \varepsilon' - j\varepsilon''$, where $\varepsilon' =$ *permittivity* and $\varepsilon'' =$ *loss factor*. Alternatively, we may write $\varepsilon = \varepsilon' - j\dfrac{\sigma}{\omega}$, where $\sigma =$ *conductivity*. Both lossless (real ε) and lossy (complex ε) dielectrics are considered. We make the common assumptions that the dielectric is simple (i.e., stationary, linear, and isotropic) and *non-magnetic* (i.e., $\mu = \mu_0 =$ permeability of free space).

An infinite-extent planar TE wave is now considered. The plane wave is incident on the layered dielectric at an angle θ_0 relative to the direction $\hat{\mathbf{z}}$, which is normal to all dielectric

boundaries (see Fig. 9). From the TE condition, the electric field is completely perpendicular to the plane of incidence; this is also known as being *perpendicularly* or *horizontally polarized.*

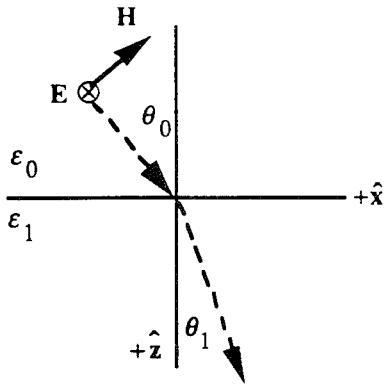


Fig. 9 TE wave geometry

The TE electric field is given by $\mathbf{E}_{TE} = \hat{\mathbf{y}} E_y e^{-j\mathbf{k}\cdot\mathbf{r}} = \hat{\mathbf{y}} E_y e^{-jk_x x} e^{-jk_z z}$. The homogeneous wave equation for a TE wave is thus: $\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial z^2} + k^2\right) E_y = 0$, obtained by substituting $\mathbf{E}_{TE}$ into

$\nabla^2 \mathbf{E} + k^2 \mathbf{E} = 0$

The solution for E_{yn} (the field in the n^{th} dielectric layer) has two wave components, one in the $+\hat{\mathbf{z}}$ direction $\{ A_n \}$ and the other in the $-\hat{\mathbf{z}}$ direction $\{ C_n \}$. We energy-normalize these components to D_n and U_n such that $A_n = D_n \left(\sqrt{\varepsilon_n} \cos\theta_n\right)^{-1/2}$ and $C_n = U_n \left(\sqrt{\varepsilon_n} \cos\theta_n\right)^{-1/2}$, where θ_n is the angle between the direction of travel of the plane wave and the $+\hat{\mathbf{z}}$ direction, in the n^{th} layer. Applying Maxwell's equations to the solution for E_{yn} to obtain H_{xn}, the tangential propagating fields are found to be

(6.1a) $$E_{yn} = \frac{1}{\left(\sqrt{\varepsilon_n} \cos\theta_n\right)^{1/2}} (D_n e^{-jk_{zn} z} + U_n e^{+jk_{zn} z}) e^{-jk_0 \sin\theta_0 x}$$

(6.1b) $$H_{xn} = \frac{-\left(\sqrt{\varepsilon_n} \cos\theta_n\right)^{1/2}}{\eta_0} (D_n e^{-jk_{zn} z} - U_n e^{+jk_{zn} z}) e^{-jk_0 \sin\theta_0 x}$$

where $\eta_0 = \sqrt{\frac{\mu_0}{\varepsilon_0}} = 377\Omega$ is the intrinsic impedance of free space, and the propagation constant $\{ k_n \}$ for a layer is related to its components $\{ k_{xn},\ k_{zn} \}$ by $k_{zn} = k_n \cos\theta_n$ and (using Snell's law) $k_{xn} = k_n \sin\theta_n = k_0 \sin\theta_0 = \omega\sqrt{\mu_0\varepsilon_0}\sin\theta_0$. The physical relationship between these components and the dielectric boundaries is shown in Fig. 10.

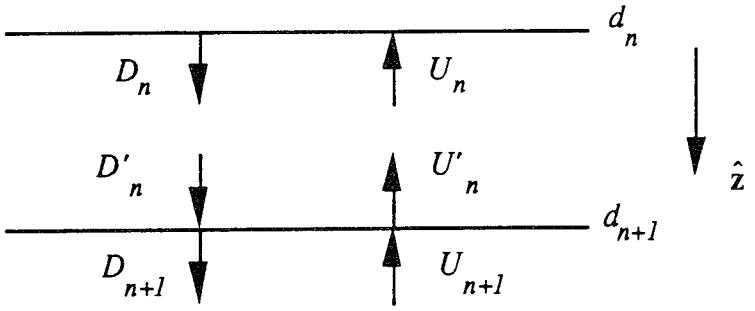


Fig. 10 Wave definitions in layered dielectrics

The goal at present is to find the relationship between the fields $\{ D_n,\ U_n \}$ in one dielectric layer and those $\{ D_{n+1},\ U_{n+1} \}$ in the next.

6.2. Formulation.

6.2A. Interface Effect. From Faraday's and Gauss's laws, at the interface $z = d_{n+1}$, we have continuity of tangential electric and magnetic fields, i.e., $E'_{yn} = E_{y(n+1)}$ and $H'_{xn} = H_{x(n+1)}$. Thus the field components $\{ D_{n+1},\ U_{n+1} \}$ just below the boundary are related to the components $\{ D'_n,\ U'_n \}$ just above the boundary by

$$\begin{bmatrix} D_{n+1}e^{-jk_{z(n+1)}d_{n+1}} \\ U_{n+1}e^{+jk_{z(n+1)}d_{n+1}} \end{bmatrix} = \frac{1}{\sqrt{1-R_{n,n+1}^2}} \begin{bmatrix} 1 & -R_{n,n+1} \\ -R_{n,n+1} & 1 \end{bmatrix} \begin{bmatrix} D'_n e^{-jk_{zn}d_{n+1}} \\ U'_n e^{+jk_{zn}d_{n+1}} \end{bmatrix}; \tag{6.2a}$$

$$R_{n,n+1} = \frac{\sqrt{\varepsilon_n}\cos\theta_n - \sqrt{\varepsilon_{n+1}}\cos\theta_{n+1}}{\sqrt{\varepsilon_n}\cos\theta_n + \sqrt{\varepsilon_{n+1}}\cos\theta_{n+1}}. \tag{6.2b}$$

At any interface of dissimilar dielectrics, both components of a propagating electromagnetic wave encounter impedance mismatches. At the interface a portion of the incident wave will be reflected, while the remainder will be transmitted. The portion reflected is determined by the TE *Fresnel reflection coefficient* $\{ R_{n,n+1} \}$.

6.2B. Homogeneous Layer Effect. Within a homogeneous dielectric layer, the reflection coefficient is zero, so that the relation between the waves $\{ D'_n,\ U'_n \}$ at the bottom of the n^{th} layer and those $\{ D_n,\ U_n \}$ at the top of the n^{th} layer is as follows. Let $\Delta_n = d_{n+1} - d_n$ be the thickness of the n^{th} layer and jk_n be the propagation constant: $jk_n = jk_0\sqrt{\varepsilon_n} = j\beta_n + \alpha_n$ is the complex sum of phase $\{ \beta_n = k_0 \operatorname{Re}\sqrt{\varepsilon_n} \}$ and attenuation $\{ \alpha_n = k_0 \left|\operatorname{Im}\sqrt{\varepsilon_n}\right| \}$ constants. Then

$$\begin{bmatrix} D'_n e^{-jk_{zn} d_{n+1}} \\ U'_n e^{+jk_{zn} d_{n+1}} \end{bmatrix} = \begin{bmatrix} e^{-\alpha_n \Delta_n \sec\theta_n} & 0 \\ 0 & e^{+\alpha_n \Delta_n \sec\theta_n} \end{bmatrix} \begin{bmatrix} e^{-j\beta_n \Delta_n \sec\theta_n} & 0 \\ 0 & e^{+j\beta_n \Delta_n \sec\theta_n} \end{bmatrix} \begin{bmatrix} D_n e^{-jk_{zn} d_n} \\ U_n e^{+jk_{zn} d_n} \end{bmatrix} \tag{6.3}$$

6.2C. Combined Effect. From phase continuity of tangential components at a boundary, we have $e^{-jk_{x(n+1)} d_{n+1}} = e^{-j\beta_{x(n+1)} d_{n+1}} e^{-\alpha_{n+1} d_{n+1}} = e^{-j\beta_{x(n+1)} d_{n+1}} e^{-\prod_{i=0}^{n} \alpha_i \Delta_i \sec\theta_i}$, where $e^{-\prod_{i=0}^{n} \alpha_i \Delta_i \sec\theta_i}$ is the cumulative loss factor $\{ L_n \}$. Defining the phasor wave components $\mathbf{D}_n = D_n e^{-j\beta_{zn} d_n}$ and $\mathbf{U}_n = U_n e^{+j\beta_{zn} d_n}$, we again find the relationship between phasor components $\{ \mathbf{D}_n,\ \mathbf{U}_n \}$ and $\{ \mathbf{D}_{n+1},\ \mathbf{U}_{n+1} \}$ given in terms of a frequency-dependent *layer propagation* matrix $\{ \mathbf{F}_{n,n+1}(\omega) \}$:

$$\begin{bmatrix} \mathbf{D}_{n+1} \\ \mathbf{U}_{n+1} \end{bmatrix} = \mathbf{F}_{n,n+1}(\omega) \begin{bmatrix} \mathbf{D}_n \\ \mathbf{U}_n \end{bmatrix} = \Sigma_{n,n+1}(\omega) \Phi_{n,n+1}(\omega) \begin{bmatrix} \mathbf{D}_n \\ \mathbf{U}_n \end{bmatrix}; \tag{6.4a}$$

with *transfer* matrix

$$\Sigma_{n,n+1}(\omega) = \frac{1}{\sqrt{1 - R_{n,n+1}^2}} \begin{bmatrix} 1 & -R_{n,n+1} e^{+2\sum_{m=0}^{n} \alpha_m \Delta_m \sec\theta_m} \\ -R_{n,n+1} e^{-2\sum_{m=0}^{n} \alpha_m \Delta_m \sec\theta_m} & 1 \end{bmatrix}; \tag{6.4b}$$

and a *time-delay* matrix

(6.4c) $$\Phi_{n,n+1}(\omega) = \begin{bmatrix} e^{-j\beta_n \Delta_n \sec\theta_n} & 0 \\ 0 & e^{+j\beta_n \Delta_n \sec\theta_n} \end{bmatrix}.$$

6.2D. Discrete-time Formulation. If we make the assumptions that the dielectrics are materials low in loss $\{\omega\varepsilon' >> \sigma\}$ and relatively independent of frequency $\{\varepsilon'(\omega) \approx \varepsilon'\}$, the following approximation can be made:

(6.5) $$j\omega\sqrt{\mu_0\varepsilon_0}\sqrt{\varepsilon_n} = j\beta_n + \alpha_n \approx \frac{j\omega}{v_n} + \frac{\eta_0\sigma_n}{2\sqrt{\varepsilon'_n}} \Rightarrow \beta_n \approx \frac{\omega}{v_n} \text{ and } \alpha_n \approx \frac{\eta_0\sigma_n}{2\sqrt{\varepsilon'_n}}.$$

We again introduce the frequency-independent transfer matrix coefficients $\{r_{n,n+1},\ s_{n,n+1}\}$ as

(6.6a) $$r_{n,n+1} = R_{n,n+1} e^{-2\sum_{m=0}^{n} \alpha_m \Delta_m \sec\theta_m};$$ (6.6b) $$s_{n,n+1} = R_{n,n+1} e^{+2\sum_{m=0}^{n} \alpha_m \Delta_m \sec\theta_m}.$$

Note that both the phase constant $\{\beta_n\}$ and traveled distance $\{\Delta_n \sec\theta_n\}$ in a layer may be written in terms of the phase velocity $\{v_n\}$ for the layer, i.e., $\beta_n \approx \frac{\omega}{v_n} = \omega\sqrt{\mu_0\varepsilon_0}\sqrt{\varepsilon'_n}$ and $\Delta_n \sec\theta_n = v_n n_n \Delta$ [16]. We again assume that travel-time through each layer is an integer $\{n_n\}$ multiple of a small time-increment $\{\Delta\}$. Then the forward layer propagation matrix can again be expressed using the z-transform as

(6.7) $$F_{n,n+1}(z) = \frac{1}{\sqrt{1 - r_{n,n+1}s_{n,n+1}}} \begin{bmatrix} 1 & -s_{n,n+1} \\ -r_{n,n+1} & 1 \end{bmatrix} \begin{bmatrix} z^{-n_n} & 0 \\ 0 & z^{n_n} \end{bmatrix} = \Sigma_{n,n+1}\Phi_{n,n+1}(z).$$

The asymptotic analysis of Section 5.3 can be applied to electromagnetic wave propagation as well, and will not be repeated here.

7. Numerical Example. We provide a simple, illustrative numerical example. The goal is to reconstruct the unknown lossy three-layer dielectric in free space shown in Fig. 11.

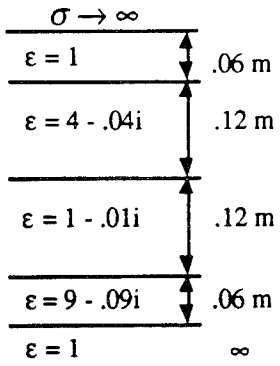


Fig. 11 Layered dielectric model

A video pulse system operating at 10 GHz is simulated. The values of permittivity $\{\varepsilon_n'\}$ and thickness $\{\Delta_n\}$ were chosen so that all returns from normal probing occur at an integer multiple of .05 nsec (of course, a smaller value could also be used). The loss factors $\{\varepsilon_n''\}$ were chosen so that the layers would substantially attenuate the signals, but small enough such that the assumption $\varepsilon_n' >> \varepsilon_n''$ is valid. The time-line bounce diagram for normal probing of this model is shown in Fig. 12.

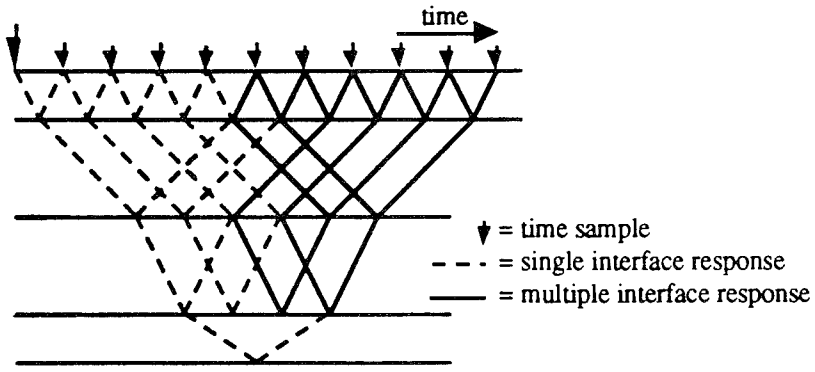


Fig. 12 Bounce diagram for normal probing

We calculate the attenuation coefficient for each layer as $\alpha_n = k_0 \left|\mathrm{Im}\sqrt{\varepsilon_n}\right| = \frac{2\pi f}{c}\left|\mathrm{Im}\sqrt{\varepsilon_n}\right|$, where $\varepsilon_n = \varepsilon_n' - i\varepsilon_n''$, $f = 10$ GHz and $c = 3\cdot 10^8 \frac{m}{s}$ (the speed of light in free space). The layer parameters are given in Table 1.

Table 1 Lossy model parameters

Layer (n)	ε_n'	Δ_n (m)	t_n (nsec)	ε''	α_n	$e^{-2\alpha_n\Delta_n}$
free space (0)	1	.06	.40	0	0	1.00
(1)	4	.12	1.60	.04	2.09	.605
(2)	1	.12	.80	.01	1.05	.778
(3)	9	.06	1.20	.09	3.14	.686
free space (4)	1	∞	∞	0	0	1.00

The reflection coefficients $\{R_{n,n+1}\}$ (calculated using (6.4b)), loss factor $\{L_n\}$ for each interface (the cumulative loss of the preceding layers, as given in Section 6.3C), and transfer coefficients $\{\mathbf{r}_{n,n+1}, \mathbf{s}_{n,n+1}\}$ for each interface of the lossy model (computed using (6.6)) are all shown in Table 2.

Table 2 Lossy model transfer coefficients

Interface (n, $n+1$)	$R_{n,n+1}$	L_n	$\mathbf{r}_{n,n+1}$	$\mathbf{s}_{n,n+1}$
0, 1	-.333	1.00	-.333	-.333
1, 2	+.333	.605	+.202	+.551
2, 3	-.500	.471	-.235	-1.063
3, 4	+.500	.323	+.161	+1.549

The actual $\{r(n)\}$ and adjoint $\{\tilde{r}(n)\}$ system free-surface reflection responses for the lossy model, computed using (3.7), are shown in Fig. 13a and Fig. 13b, respectively.

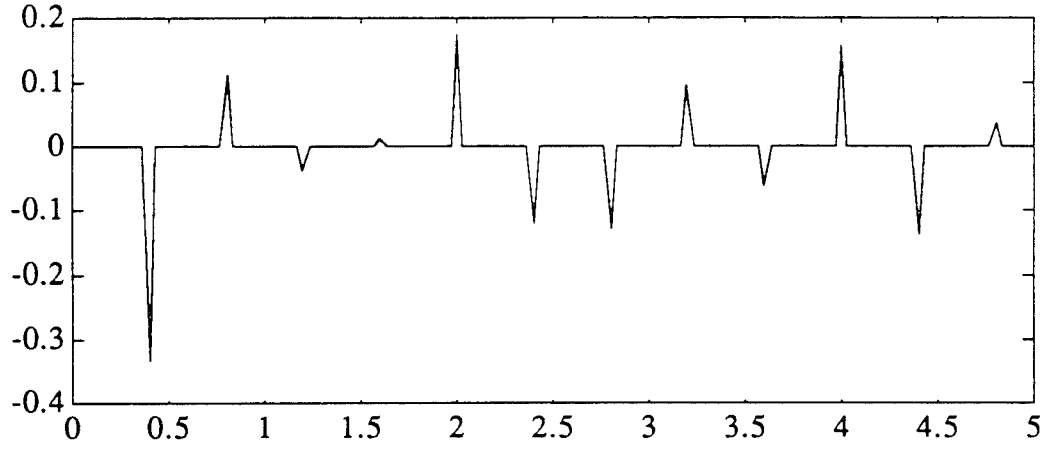


Fig. 13a Actual system reflection response vs. time (nsec)

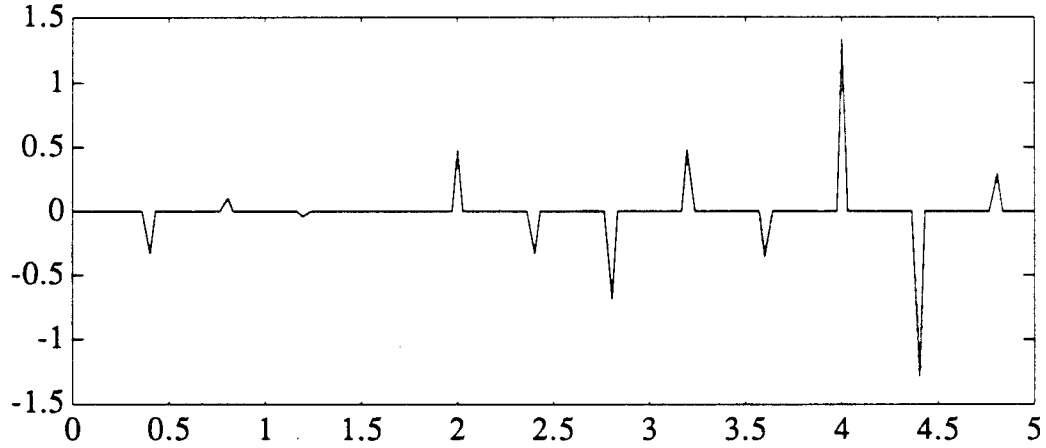


Fig. 13b Adjoint system reflection response vs. time (nsec)

The sampled responses $r(n)$ and $\tilde{r}(n)$ are assembled into an asymmetric Toeplitz system, which is solved using the asymmetric Levinson algorithm. This produces two sets of transfer coefficients as functions of elapsed time: one for the actual system $\{\mathbf{r}\}$; and one for the adjoint system $\{\mathbf{s}\}$. The transfer coefficients $\{\mathbf{r}_{n,n+1},\ \mathbf{s}_{n,n+1}\}$ for the lossy model are shown in Fig. 14a and Fig. 14b, respectively. Comparing Fig. 13 and Fig. 14 shows the significance of multiple reflections.

From these values we can obtain $\mathbf{r}_{n,n+1}$ and $\mathbf{s}_{n,n+1}$. Hence, the lossy medium's parameters $\{\varepsilon_n,\ \Delta_n,$ and $\alpha_n\ \forall n\}$ can be completely reconstructed using (6.26), (6.5), and (6.6). The reconstructed parameters are identical to the actual parameters in Table 1, and are not repeated here.

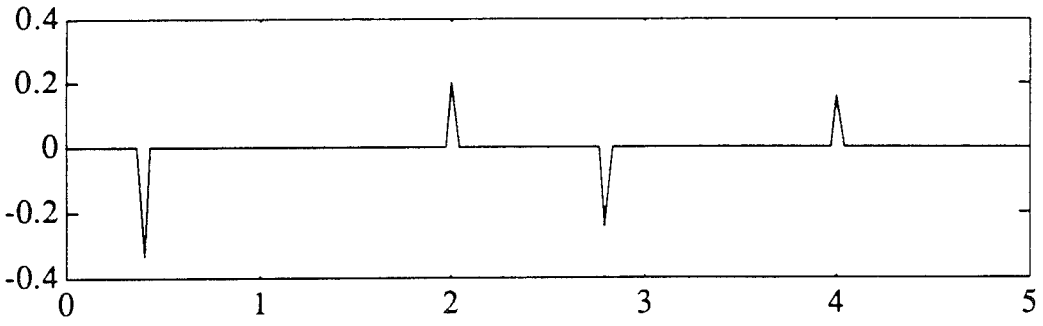


Fig. 14a Transfer coefficients $\{\mathbf{r}\}$ *for actual system vs. time (nsec)*

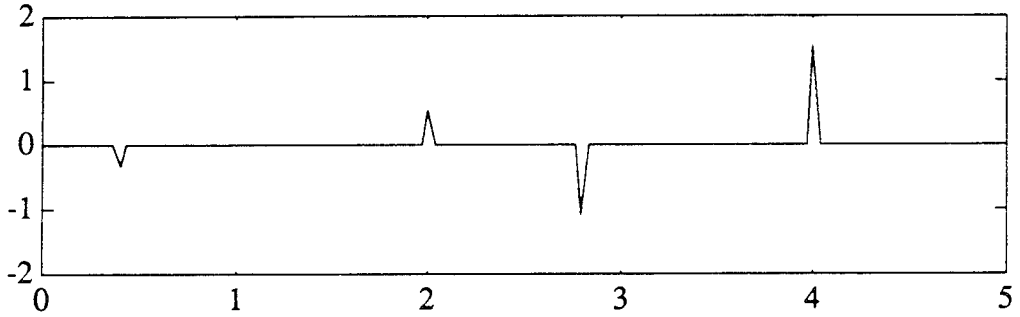


Fig. 14b Transfer coefficients $\{\mathbf{s}\}$ *for adjoint system vs. time (nsec)*

8. Conclusion. We have extended the development of a DSP theory for forward and inverse scattering from discrete lossless systems to discrete lossy systems. Discrete lossy systems are in fact more useful for modeling many real-world scattering and inverse scattering problems. The advantage of the DSP formulation is that the signal processing can be carried out exactly, whereas continuous-time problems must be discretized, and effects such as transmission losses and dispersion do not appear. The development of inverse scattering for discrete lossy systems was presented. Major contributions of this paper include: (1) a discrete-time formulation of scattering for lossy systems; (2) an explicit derivation of the asymmetric Toeplitz system for inverse scattering for a discrete lossy system; (3) explicit formulations of the discrete lossy

transmission line and layered dielectric medium problems as discrete asymmetric two-component wave systems; (4) discussions of the significance of data feasibility for lossy inverse scattering problems; and (5) an asymptotic frequency analysis of the lossy transmission line, and a discussion of the relative significance of absorption and dispersion effects. The latter analysis demonstrates that for real-world problems with high-frequency pulses, our DSP formulation models the significant features of the problems.

Acknowledgments. The work of the first author was supported by a fellowship from Hughes Aircraft Company. The work of the second author was supported by the Office of Naval Research under grant #N00014-90-J-1897.

References.

[1] A. M. Bruckstein and T. Kailath, *Inverse scattering for discrete transmission-line models*, SIAM Review, Vol. 29, No. 3, Sept. 1987

[2] K. Case and M. Kac, *A discrete version of the inverse scattering problem*, J. Math. Phys., Vol. 14, No. 5, May 1973, 594-603.

[3] J. Berryman and R. Greene, *Discrete inverse methods for elastic waves in layered media*, Geophysics, Vol. 45, No. 2, Feb. 1980, 213-33.

[4] P. Smith, *Digital realisation of forward and inverse models of electromagnetic scattering*, Electronics Letters, Vol. 25, 1989, 816-7.

[5] A. E. Yagle and B. C. Levy, *The Schur algorithm and its applications*, Acta Applicandae Mathematicae,3, 255-84, 1985.

[6] A. M. Bruckstein, T. Kailath, and B. C. Levy, *Differential methods in inverse scattering*, SIAM J. Appl. Math, Vol. 45, 312-35, 1985.

[7] V. Bogorodsky, C. Bentley, and P. Gudmandsen, *Radioglaciology*, D. Reidel Publishing Co., 1985.

[8] D. Goeckel, K. Webb, N. Gallagher, *Massively parallel iterative determination of stratified dielectric parameters from scattered-field measurements*, J. Opt. Soc. Am. A., Vol. 10, No. 5, May 1993.

[9] *IEEE Transactions on Circuits and Systems*, Nov. 1992 (entire issue).

[10] K. Winick, *Effective-index method and coupled-mode theory for almost-periodic waveguide gratings: a comparision*, Applied Optics, Vol. 31, No.6, pp. 757-64, Feb. 1992.

[11] G. Kunetz and I. d'Erceville, *Sur certains proprietes d'une onde acoustiuqe plane de compression dans un milieu stratifie*, Annales de Geophysique, Vol. 18, 1962, pp. 351-9.

[12] A. M. Bruckstein, I. Koltracht, and T. Kailath, *Inverse scattering with noisy data*, SIAM J. Sci. Stat. Comput., Vol. 7, No. 4, pp. 1331-49, Oct. 1986.

[13] G. Carayannis, N. Kalouptsidis, and D. G. Manolakis, *Fast recursive algorithms for a class of linear equations*, IEEE Trans. Acoust., Speech, Sig. Proc., Vol. ASSP-30, pp. 227-39, 1982.

[14] M. Jaulent, *Inverse scattering problem for LRCG transmission lines*, J. Math. Phys., Vol. 23, 2286-90, 1982.

[15] J. Frolik and A. E.Yagle, *Reconstruction of multi-layered lossy dielectrics from plane-wave impulse responses at two angles of incidence*, submitted to IEEE Trans. Geosci. and Rem. Sensing, June 1993.

[16] R. Harrington, *Time-Harmonic Electromagnetic Fields*, McGraw-Hill, 1961.

APPENDIX E2

J. Frolik and A.E. Yagle, "Reconstruction of Multi-Layered Lossy Dielectrics from Plane Wave Impulse Responses at Two Angles of Incidence," submitted to IEEE Trans. Geosci. and Rem. Sensing, June 1993.

The problem posed in Appendix E1 is solved using plane wave reflection response at two angles of incidence, rather than reflection and transmission data (the latter would not be available in remote sensing applications). This includes a novel seni-iterative use of layer stripping. Numerical examples on reconstructing a glacial ice shelf from radar reflections demonstrate the significance of modelling multiple reflections and losses.

Reconstruction of Multilayered Lossy Dielectrics from Plane Wave Impulse Responses at Two Angles of Incidence

Jeffrey L. Frolik, *Student Member, IEEE*, and Andrew E. Yagle, *Fellow, IEEE*

Abstract— **Motivated by the radioglaciology inverse problem, we present new algorithms for reconstructing a lossy, stratified dielectric from its impulsive plane wave reflection responses at two different angles of incidence. Novel features of these algorithms include: 1) a digital signal processing formulation that does not require discretization of continuous equations; 2) use of the asymmetric Levinson algorithm for rapid solution of the forward and inverse problems; 3) a novel use of layer stripping ideas, featuring iteration between the forward and inverse problems, with each iteration recursively determining another layer of the medium; and 4) another recursive algorithm for determining the bottom lossy half-space from reflection data only. Numerical examples illustrate the new algorithms on the reconstruction of a synthetic but realistic layered ice shelf from both noiseless and noisy radar reflection data.**

I. Introduction

THE electromagnetic plane wave inverse scattering problem is to compute the electrical parameters (e.g., permittivity, attenuation constant) of a layered dielectric from scattering data (e.g., reflection response to an impulsive plane wave). Previous methods can be classified into two categories: 1) those based on integral-inverse algorithms (e.g., Gel'fand-Levitan-Marchenko and parameter optimization techniques); and 2) those based on differential-inverse algorithms (e.g., layer stripping techniques). Comprehensive reviews of previous work can be found in [1], [2] and their associated references.

A few methods applicable to the case of lossy dielectrics are as follows: *Iterative* methods for simultaneously reconstructing permittivity and conductivity profiles using the distorted Born approximation have been applied to a variety of geometries [20]. However, this method is suitable only for smoothly varying media and would not be appropriate for investigating stratified media having dissimilar layers. Time domain methods have been used to reconstruct conductivity profiles as presented in [21] and its references. However, this work has assumed that the permittivity profile is known *a priori*. A third alternative method of reconstructing these profiles based on the Fourier transform of surface impedance data was presented in [3]. This method assumes that the layer travel times are all equal, so that the scaled discrete-time Fourier transform of the reflection response will be periodic with period in Hertz equal to the reciprocal of the travel time. If the reflection response is measured over one period at very high frequencies, the variation of medium response with frequency can be neglected. However, this method will break down completely unless the layer thicknesses are all equal, and even then it will require measurement at impractical high frequencies. For a travel time of 0.5 nsec, which we use in the examples of this paper, a 1% variation of loss would require measurements over the range of 200–202 GHz; such frequencies would not penetrate far anyway. Although we assume the medium is probed with an impulsive plane wave, in fact any fast pulse of duration less than the travel time between interfaces may be used.

Methods presented in this paper are *recursive* algorithms for reconstructing *both* permittivity and attenuation profiles in layered media that find their origins in the differential-inverse class. This paper differs from previous work in four important respects: 1) we present algorithms for reconstructing a *lossy* dielectric, consisting of a stack of discrete, homogeneous, and *absorbing* layers, from only the *reflection* responses to impulsive plane waves at two different angles of incidence; 2) the algorithms can be implemented in terms of digital signal processing (DSP) techniques and algorithms—no discretization of continuous equations is required; 3) a novel, *recursive* use of layer stripping ideas, in which lossy layers of the stratified dielectric are recursively reconstructed by iterating between the forward and inverse problems, both of which can be solved rapidly by running the asymmetric Levinson algorithm forward and backward; and 4) the first algorithm that can reconstruct the bottom, lossy half-space entirely from the reflection response data.

It is important to recognize that reconstruction of attenuation profiles differs significantly from reconstruction of layer parameters such as permittivity. The latter can be discerned from the interface reflection coefficient, but the former does not appear in the reflection coefficient. The attenuation of a layer changes in the same way that the phase velocity does as the angle of incidence is varied. Hence, probing at more than two angles of incidence does not help. However, it does help if the data are noisy [4].

We also present numerical examples that: 1) demonstrate the significance of multiple reflections and absorption on reconstruction; 2) demonstrate that the new, proposed algorithms work properly; 3) demonstrate the significance of the recursive

Manuscript received July 7, 1993; revised February 22, 1994. This work was supported in part by the Office of Naval Research under Grant #N00014-90-J-1897 and in part by a fellowship from the Hughes Aircraft Company.

The authors are with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109-2122 USA.

IEEE Log Number 9408486.

0196–2892/95$04.00 © 1995 IEEE

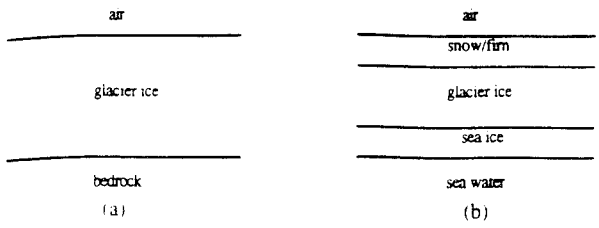


Fig. 1. Glacier models (a) single layer model, (b) multi-layer model.

algorithm in reconstructing a lossy layered dielectric, especially an unknown lossy lower half-space; and 4) demonstrate the effectiveness of the routines in the presence of noise.

This paper is organized as follows: Section II discusses the radioglaciology inverse problem which motivates this work (although it can also be applied to magnetotelluric exploration, FM-CW subsurface radar, and radar reflections off of layered composites, e.g., aircraft skins). Section III derives the asymmetric wave system that is used to characterize electromagnetic wave propagation in lossy, stratified dielectrics (this is discussed in more detail in [5]). Section IV shows how running the asymmetric Levinson algorithm forward and backward solves the inverse and forward problems, respectively. Section IV also presents a simpler algorithm for low-loss media. Section V shows how the algorithms of Section IV can be used recursively to reconstruct the bottom layer of a lossy layered medium. Section VI presents numerical examples that demonstrate the operation and success of the algorithms in reconstructing realistic lossy layered dielectrics, including the bottom half-space, from both noiseless and noisy data. Section VII concludes with a summary.

II. Summary of Radioglaciology

Radar remote sensing is a one-sided probing method used to investigate both naturally-occurring and man-made phenomena, e.g., soils, forest canopies, glaciers, walls, etc. At large wavelengths the boundaries between layers is effectively a planar interface, thus many of these media can be modeled as discrete multilayer lossy media. For example, a vegetation canopy can be modeled as two layers of vegetation over a ground surface. The top layer of vegetation contains branches, leaves, needles, etc., and the bottom contains tree trunks [6].

Another example is glaciers, which are of interest in numerous disciplines, e.g., meteorology, climatology, geophysics, etc. Glacial structure lends itself well to the problem of inverse scattering for a discrete lossy systems, for they may be modeled as: 1) a single homogeneous layer of infinite horizontal extent; or as 2) a multilayer model of homogeneous layers having parallel interfaces; both models being terminated by a half-space, e.g., rock or water. When the glacier is modeled as a single homogeneous slab [Fig. 1(a)], the reflection due to the glacier/bedrock interface has strength determined by the power reflection coefficient (PRC) and the mean dielectric absorption ($\bar{B}$) of the glacier—the total radar attenuation in dB in a glacier of thickness (z) is $2\bar{B}z + \text{PRC}$ [7].

However, glaciers tend to have intervening layers of different density, structure, and impedance characteristics, which also attenuate the probing signal and produce multiple reflections. These intervening layers may be firn (pack snow), sea ice, ice lenses (higher-density frozen rainwater), or moraine. Ignoring the intervening layers and their multiple reflections leads to poor estimation of the glacier's actual properties, e.g., thickness. Thus the multi-layer glacier model [Fig. 1(b)] characterizes the physical system more effectively. We will show that the multiple reflections provide additional information and can be used to reconstruct the thickness, wave speed, and loss of all the layers of the medium.

Historically, glacial stratification has been measured using seismic sounding. In the late 1960's an alternative to seismic sounding was developed using pulse (or impulse) radar. This method is known today as *radio echo sounding* (RES). The absorption of radio waves by ice is sufficiently low to make RES a feasible probing technique. In fact, the most widespread application of RES is the measurement of glacier thickness [8]. RES has been found to produce data which complements that of seismic exploration. For example, seismic wave velocities increase with density, while electromagnetic wave speed is independent of density. Also, the variation in electromagnetic loss with temperature is greater than the variation of acoustic loss. Thus RES is becoming an increasingly popular method of glacier investigation.

RES systems operate at lower frequencies (60–600 MHz) than conventional radar. This enables greater penetration than is possible at higher frequencies [9]. In addition, given the longer wavelengths, interface roughness and slope are less significant, so the layered model is more accurate. Early RES of glaciers was performed at a center frequency of 620 MHz [10]. More recent work has used frequencies from 60 MHz to 440 MHz [11] and even as low as 1 MHz for increased penetration [12].

Pulse radar operates on the simple principle that pulses are not only partially *absorbed* within each ice layer, but also partially *reflected* when an interface of dissimilar layers is encountered. The time it takes a pulse to travel to and from an interface is $T_n = \frac{2R_n}{v_n}$, where R_n is the depth of the interface and v_n is the medium's velocity of electromagnetic wave propagation (*wave velocity*). Due to the complicated layer structure of glaciers, *a priori* knowledge of the medium's wave velocity $\{v_n\}$ is limited to the first layer. Yet the reconstruction depends on knowing for v_n *each* layer. Several techniques are used in practice to measure v_n in glaciers, e.g., interferometry technique, sounding next to drill holes of known depth, and oblique reflection sounding. Oblique reflection sounding is a standard technique of exploration geophysics, commonly used for seismic measurements, in which the variation of the travel-time for a bottom reflection is measured as the transmitter and receiver are separated along the surface.

We show that, by applying a discrete lossy model, one-sided probing at two angles can be used to reconstruct the parameters (wave velocity, thickness and loss) of each ice shelf layer without having *a priori* knowledge of the wave speed for each layer. Previous inverse scattering techniques have also required *a priori* knowledge of the bottom half-space, or a perfectly reflecting surface at the final interface, to perform reconstruction [2]. We show that complete reconstruction can be performed without these constraints by presenting a

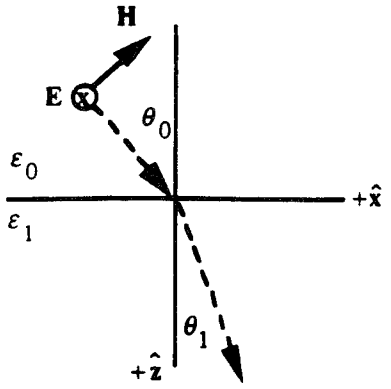


Fig. 2. TE wave geometry.

numerical example, in which oblique probing of an ice shelf [Fig. 1(b)] is simulated.

III. Asymmetric Wave System for Lossy Media

We quickly review the derivation of the asymmetric wave system for electromagnetic wave propagation in lossy, stratified dielectrics [5]. Equation (8) that follows will be vital in subsequent sections.

A. Basic Equations

The *time-harmonic* form of a propagating electromagnetic wave consists of orthogonal electric $\{\mathbf{E} = Ee^{-j\mathbf{k}\cdot\mathbf{r}}\}$ and magnetic $\{\mathbf{H} = He^{-j\mathbf{k}\cdot\mathbf{r}}\}$ field vectors having amplitudes $\{E = Ee^{j\omega t}\}$ and $\{H = He^{j\omega t}\}$, respectively, and frequency $\{\omega\}$. The propagating waves $\{\mathbf{E}, \mathbf{H}\}$ each have a temporal $\{\omega t\}$ and spatial $\{\mathbf{k}\cdot\mathbf{r}\}$ component, where $\mathbf{k} = k_x\hat{\mathbf{x}} + k_y\hat{\mathbf{y}} + k_z\hat{\mathbf{z}}$ is the propagation vector and $\mathbf{r} = x\hat{\mathbf{x}} + y\hat{\mathbf{y}} + z\hat{\mathbf{z}}$ is the displacement vector. The magnitude of the propagation vector is the *wavenumber* or *propagation constant* $k = |\mathbf{k}| = \omega\sqrt{\mu\varepsilon}$; we note that $k = \sqrt{k_x^2 + k_y^2 + k_z^2}$.

In general, the dielectric constant $\{\varepsilon^r = \frac{\varepsilon}{\varepsilon_0}\}$ is complex and may be expressed as $\varepsilon^r = \varepsilon' - j\varepsilon''$, where $\varepsilon' =$ *relative permittivity* and $\varepsilon'' =$ *loss factor* (ε_0 = permittivity of free space). Alternatively, we may write $\varepsilon^r = \varepsilon' - j\frac{\sigma}{\omega}$, where $\sigma =$ *conductivity*. A lossless and lossy medium are specified by real and complex ε^r, respectively. We make the common assumptions that the dielectric is *simple* (i.e., stationary, linear, and isotropic) and *nonmagnetic* (i.e., $\mu = \mu_0 =$ permeability of free space).

An infinite-extent planar TE wave is considered. The plane wave is incident on the layered dielectric at an angle θ_0 relative to the direction $\hat{\mathbf{z}}$, which is normal to all dielectric boundaries (Fig. 2). The layered dielectric is defined as a stack of parallel dielectric slabs having infinite lateral extent. From the TE condition, the electric field lies completely perpendicular to the plane of incidence; this is also known as being *perpendicularly* or *horizontally polarized*.

The TE electric field is given by $\mathbf{E} = \hat{\mathbf{y}}E_y e^{-j\mathbf{k}\cdot\mathbf{r}} = \hat{\mathbf{y}}E_y e^{-jk_x x}\, e^{-jk_z z}$ and thus the homogeneous wave equation becomes

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial z^2} + k^2\right)E_y = 0. \tag{1}$$

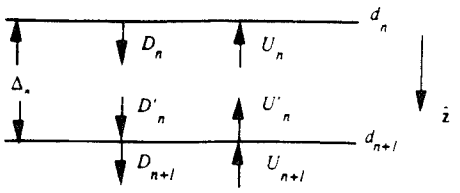


Fig. 3. Wave definitions in layered dielectrics.

Solving (1) in the nth dielectric layer results in a solution for E_{yn} of two energy-normalized wave components, one in the $+\hat{\mathbf{z}}$ direction $\{D_n\}$ and the other in the $-\hat{\mathbf{z}}$ direction $\{U_n\}$ (see Fig. 3).

The waves D_n and U_n in the nth layer are defined from the tangential fields E_{yn} and H_{xn} [19] as

$$D_n = \frac{1}{2}\left[\left(\sqrt{\varepsilon_n^r}\cos\theta_n\right)^{\frac{1}{2}}E_{yn} - \left(\sqrt{\varepsilon_n^r}\cos\theta_n\right)^{-\frac{1}{2}}\eta_0 H_{xn}\right] \times e^{+jk_{xn}x} \tag{2a}$$

$$U_n = \frac{1}{2}\left[\left(\sqrt{\varepsilon_n^r}\cos\theta_n\right)^{\frac{1}{2}}E_{yn} + \left(\sqrt{\varepsilon_n^r}\cos\theta_n\right)^{-\frac{1}{2}}\eta_0 H_{xn}\right] \times e^{+jk_{xn}x} \tag{2b}$$

where $\eta_0 = \sqrt{\frac{\mu_0}{\varepsilon_0}} = 377\Omega$ is the intrinsic impedance of free space and $k_n = \omega\sqrt{\mu_0\varepsilon_n}$, $k_{xn} = k_n\sin\theta_n = k_0\sin\theta_0$ (using Snell's law) and $k_{zn} = k_n\cos\theta_n$ or equivalently, $k_{zn} = (\omega^2\mu_0\varepsilon_n - k_0^2\sin^2\theta_0)^{\frac{1}{2}}$ (using the consistency condition). Note that the x-dependence is entirely contained in $e^{+jk_{xn}x}$, which is why plane wave probing at nonnormal incidence can be reformulated as plane wave probing at normal incidence.

B. Interface Effects

We now find the relationship between the waves $\{D_n, U_n\}$ in one dielectric layer and those $\{D_{n+1}, U_{n+1}\}$ in the next. By the continuity of tangential electric and magnetic fields at the interface ($z = d_{n+1}$), the field components $\{D_{n+1}, U_{n+1}\}$ just below the boundary are related to components $\{D'_n, U'_n\}$ just above the boundary (see Fig. 3) by

$$\begin{bmatrix} D_{n+1} \\ U_{n+1} \end{bmatrix} = \frac{1}{\sqrt{1 - R_{n,n+1}^2}}\begin{bmatrix} 1 & -R_{n,n+1} \\ -R_{n,n+1} & 1 \end{bmatrix}\begin{bmatrix} D'_n \\ U'_n \end{bmatrix}; \tag{3a}$$

$$R_{n,n+1} = \frac{\sqrt{\varepsilon_n}\cos\theta_n - \sqrt{\varepsilon_{n+1}}\cos\theta_{n+1}}{\sqrt{\varepsilon_n}\cos\theta_n + \sqrt{\varepsilon_{n+1}}\cos\theta_{n+1}}; \tag{3b}$$

$$R^v_{n,n+1} = \frac{\sqrt{\varepsilon_{n+1}}\cos\theta_n - \sqrt{\varepsilon_n}\cos\theta_{n+1}}{\sqrt{\varepsilon_{n+1}}\cos\theta_n + \sqrt{\varepsilon_n}\cos\theta_{n+1}}. \tag{3c}$$

At any interface of dissimilar dielectrics, both components of a propagating electromagnetic wave encounter impedance mismatches. At the interface a portion of the incident wave will be reflected, while the remainder will be transmitted. The portion reflected is determined by the TE *Fresnel reflection coefficient* $\{R_{n,n+1}\}$ (3b); for a *vertically polarized, parallel,* or TM wave, the Fresnel reflection coefficient is $\{R^v_{n,n+1}\}$.

C. Homogeneous Layer Effects

Within a homogeneous dielectric layer, the reflection coefficient is zero, so that the relation between the waves $\{D'_n, U'_n\}$

at the bottom of the nth layer and those $\{D_n, U_n\}$ at the top of the nth layer is given by the propagation effect through the layer, where $\Delta_n = d_{n+1} - d_n$ is the thickness of the nth layer. For a lossy layer, the propagation constant $\{k_n\}$ is complex. Specifically, the propagation constant in the $\pm\hat{\mathbf{z}}$ direction is given by $\{k_{zn} = \beta_{zn} - j\alpha_{zn}\}$ and is the complex sum of attenuation $\{\alpha_{zn}\}$ and phase $\{\beta_{zn}\}$ constants. We thus have

$$\begin{bmatrix} D_n' \\ U_n' \end{bmatrix} = \begin{bmatrix} e^{-j\beta_{zn}\Delta_n - \alpha_{zn}\Delta_n} & 0 \\ 0 & e^{+j\beta_{zn}\Delta_n + \alpha_{zn}\Delta_n} \end{bmatrix} \begin{bmatrix} D_n \\ U_n \end{bmatrix}. \tag{4}$$

D. Combined Effects

Define $\mathbf{A}_{n+1} = \sum_{i=0}^{n} \alpha_{zi}\Delta_i = \mathbf{A}_n + \alpha_{zn}\Delta_n$ and the scaled waves $\mathbf{D}_n = D_n e^{+\mathbf{A}_n}$ and $\mathbf{U}_n = U_n e^{-\mathbf{A}_n}$. Then, using the commutativity of multiplication of diagonal matrices, (3a) and (4) can be combined into

$$\begin{aligned} \begin{bmatrix} \mathbf{D}_{n+1} \\ \mathbf{U}_{n+1} \end{bmatrix} &= \begin{bmatrix} e^{+\mathbf{A}_{n+1}} & 0 \\ 0 & e^{-\mathbf{A}_{n+1}} \end{bmatrix} \\ &\quad \times \frac{1}{\sqrt{1 - R_{n,n+1}^2}} \begin{bmatrix} 1 & -R_{n,n+1} \\ -R_{n,n+1} & 1 \end{bmatrix} \\ &\quad \times \begin{bmatrix} e^{-j\beta_{zn}\Delta_n - \alpha_{zn}\Delta_n} & 0 \\ 0 & e^{+j\beta_{zn}\Delta_n + \alpha_{zn}\Delta_n} \end{bmatrix} \\ &\quad \times \begin{bmatrix} e^{-\mathbf{A}_n} & 0 \\ 0 & e^{+\mathbf{A}_n} \end{bmatrix} \begin{bmatrix} \mathbf{D}_n \\ \mathbf{U}_n \end{bmatrix} \\ &= \frac{1}{\sqrt{1 - R_{n,n+1}^2}} \\ &\quad \times \begin{bmatrix} 1 & -R_{n,n+1}e^{+2\mathbf{A}_{n+1}} \\ -R_{n,n+1}e^{-2\mathbf{A}_{n+1}} & 1 \end{bmatrix} \\ &\quad \times \begin{bmatrix} e^{-j\beta_{zn}\Delta_n} & 0 \\ 0 & e^{+j\beta_{zn}\Delta_n} \end{bmatrix} \begin{bmatrix} \mathbf{D}_n \\ \mathbf{U}_n \end{bmatrix}. \end{aligned} \tag{5}$$

The first matrix is the *transfer* matrix, the second matrix is the *time-delay matrix*, and their product is the layer matrix.

E. Discrete-Time Formulation

If we make the assumptions that the dielectrics are materials low in loss $\{|\omega\varepsilon_n| \gg \sigma_n\}$ and relatively independent of frequency $\{\varepsilon_n(\omega) \approx \varepsilon_n\}$, the following approximation can be made

$$\begin{aligned} & j\omega\sqrt{\mu_0\varepsilon_0}\sqrt{\varepsilon_n'}\cos\theta_n \\ &\quad = j\beta_{zn} + \alpha_{zn} \approx \left(\frac{j\omega}{v_n} + \frac{\eta_0\sigma_n}{2\sqrt{\varepsilon_n'}}\right)\cos\theta_n \\ &\quad \Rightarrow \beta_{zn} \approx \frac{\omega}{v_{zn}} \quad \text{and} \quad \alpha_{zn} \approx \frac{\eta_0\sigma_n}{2\sqrt{\varepsilon_n'}}\cos\theta_n, \end{aligned} \tag{6}$$

where $v_n = (\mu_0\varepsilon_0\varepsilon_n')^{-\frac{1}{2}}$ is the wave speed in the nth layer and $v_{zn} = v_n/\cos\theta_n$ is the phase velocity for the nth layer in the $\pm\hat{\mathbf{z}}$ direction. We introduce the following frequency-independent transfer matrix coefficients $\{\mathbf{r}_{n,n+1}, \mathbf{s}_{n,n+1}\}$:

$$\mathbf{r}_{n,n+1} = R_{n,n+1}e^{-2\mathbf{A}_{n+1}}; \tag{7a}$$

$$\mathbf{s}_{n,n+1} = R_{n,n+1}e^{+2\mathbf{A}_{n+1}}. \tag{7b}$$

Note that the product of phase constant $\{\beta_{zn}\}$ and traveled distance $\{\Delta_n\}$ may be written in terms of the phase velocity $\{v_{zn}\}$, i.e., $\beta_{zn}\Delta_n = \frac{\omega}{v_{zn}}\Delta_n = \omega T_{zn}$ [13], where T_{zn} is the travel-time in the $\{\pm\hat{\mathbf{z}}\}$ direction for the nth layer. We also assume that layer travel distances $\{\Delta_n\}$ are such that $\frac{\Delta_n}{v_{zn}} = T_{zn} = n_n T$, i.e., the travel-time $\{T_{zn}\}$ is an integer $\{n_n\}$ multiple of a very small time increment $\{T\}$ (in the example of Section VI we use 0.5 nsec). Replacing the Fourier kernel in (5) with the z-transform kernel, i.e., $e^{j\omega} \Rightarrow z$, and scaling $T = 1$ without loss of generality, we obtain the *discrete-time* expression of the layer matrix:

$$\begin{aligned} \begin{bmatrix} \mathbf{D}_{n+1}(z) \\ \mathbf{U}_{n+1}(z) \end{bmatrix} &= \frac{1}{\sqrt{1 - \mathbf{r}_{n,n+1}\mathbf{s}_{n,n+1}}} \begin{bmatrix} 1 & -\mathbf{s}_{n,n-1} \\ -\mathbf{r}_{n,n+1} & 1 \end{bmatrix} \\ &\quad \times \begin{bmatrix} z^{-n_n} & 0 \\ 0 & z^{n_n} \end{bmatrix} \begin{bmatrix} \mathbf{D}_n(z) \\ \mathbf{U}_n(z) \end{bmatrix}. \end{aligned} \tag{8}$$

The significance of (8) is twofold. First, the unequal transfer coefficients $\{\mathbf{r}_{n,n+1}, \mathbf{s}_{n,n+1}\}$ model losses in the medium. Second, the explicitly discrete-time formulation we have provided (emphasized by our use of the z-transform) means that digital signal processing (DSP) techniques can be used to obtain exact answers. For example, we are *not* using the discrete Fourier transform (DFT) as an approximation to the continuous Fourier transform, as is often the case. The DFT will give the exactly correct answer for our problem, since the measured data and the assumed model of the medium are both discrete. Our explicitly discrete formulation solves the problem of reconstructing a discrete layered medium exactly using DSP techniques for their own sakes, not as approximations to continuous equations. Note that (8) is analogous to equations describing asymmetric lattice filters in DSP theory.

F. Inverse Scattering

Let $r(n)$ be the discrete-time impulse reflection response of the dielectric medium, which consists of a stack of layers in each of which wave propagation is described using (8). Let $\tilde{r}(n)$ be the impulse reflection response of the *adjoint medium*, a *fictitious* medium which consists of a stack of fictitious layers in each of which wave propagation is described using (8) *with* $\mathbf{r}_{n,n+1}$ *and* $\mathbf{s}_{n,n+1}$ *exchanged* (of course, $\tilde{r}(n)$ is not directly measurable, but we will never need to measure $\tilde{r}(n)$; see Section IV). In [5], it has been shown that $\mathbf{r}_{n,n+1}$ and $\mathbf{s}_{n,n+1}$ are the reflection coefficients computed when the following asymmetric Toeplitz system is solved using the asymmetric Levinson algorithm:

$$\begin{aligned} &\begin{bmatrix} 1 & r(1) & r(2) & \cdot & \cdot & r(n) \\ \tilde{r}(1) & 1 & r(1) & & & \cdot \\ \tilde{r}(2) & \tilde{r}(1) & 1 & \cdot & & \cdot \\ \cdot & & \cdot & \cdot & \cdot & \cdot \\ \cdot & & & \cdot & \cdot & r(1) \\ \tilde{r}(n) & \cdot & \cdot & \cdot & \tilde{r}(1) & 1 \end{bmatrix} \begin{bmatrix} a_n(0) & b_n(n) \\ a_n(1) & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & b_n(1) \\ a_n(n) & b_n(0) \end{bmatrix} \\ &= \begin{bmatrix} \tau_n & 0 \\ 0 & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & 0 \\ 0 & \tau_n \end{bmatrix} \end{aligned} \tag{9}$$

where the asymmetric Levinson algorithm is

TABLE I
LAYERED DIELECTRIC RECONSTRUCTION FROM TRANSFER COEFFICIENTS

Lossless Dielectric	*Lossy Dielectric*
Given: $R_{n,n+1}$, T_n Find: ε_n', Δ_n	Given: $\mathbf{r}_{n,n+1}$, $\mathbf{s}_{n,n+1}$, T_n Find: ε_n', Δ_n, α_n
Step 1. Initialize with $\varepsilon_0' = 1$, $n = 0$.	1. Initialize with $\varepsilon_0' = 1$, $L_0 = 1$, $n = 0$.
Step 2. $\Delta_n = \dfrac{T_n c}{2\sqrt{\varepsilon_n'}}$; c = *speed of light*	2. $\Delta_n = \dfrac{T_n c}{2\sqrt{\varepsilon_n'}}$
	3. $R_{n,n+1} = \operatorname{sgn}(\mathbf{r}_{n,n+1})\sqrt{\mathbf{r}_{n,n+1}\mathbf{s}_{n,n+1}}$ (since the loss factor is always positive)
Step 3. $\sqrt{\varepsilon_{n+1}'} = \dfrac{1 - R_{n,n+1}}{1 + R_{n,n+1}}\sqrt{\varepsilon_n'}$	4. $\sqrt{\varepsilon_{n+1}'} = \dfrac{1 - R_{n,n+1}}{1 + R_{n,n+1}}\sqrt{\varepsilon_n'}$
	5. $L_n = \dfrac{\mathbf{r}_{n,n+1}}{R_{n,n+1}} \Rightarrow \alpha_n = \dfrac{-1}{2\Delta_n}\ln\dfrac{L_n}{L_{n-1}}$
Step 4. Increment n, return to Step 2.	6. Increment n, return to Step 2.

Asymmetric Levinson Algorithm

Step 1: Initialize: $n = 0$, $A_0(z) = B_0(z) = 1$, *and* $\tau_0 = 1$.
Step 2:

$$\mathbf{r}_{n,n+1} = -\sum_{i=0}^{n}\frac{a_{n,i}r(n+1-i)}{\tau_n}\mathbf{s}_{n,n+1} = \sum_{i=0}^{n}\frac{b_{n,i}\tilde{r}(i+1)}{\tau_n}$$

Step 3:

$$\begin{bmatrix}A_{n+1}(z)\\ B_{n+1}(z)\end{bmatrix} = \frac{z^{+\frac{1}{2}}}{\sqrt{1-\mathbf{r}_{n,n+1}\mathbf{s}_{n,n+1}}} \times \begin{bmatrix}1 & -\mathbf{s}_{n,n+1}\\ -\mathbf{r}_{n,n+1} & 1\end{bmatrix}\begin{bmatrix}z^{-\frac{1}{2}} & 0\\ 0 & z^{+\frac{1}{2}}\end{bmatrix}\begin{bmatrix}A_n(z)\\ B_n(z)\end{bmatrix}$$

Step 4: $\tau_{n+1} = \tau_n\sqrt{1-\mathbf{r}_{n,n+1}\mathbf{s}_{n,n+1}}$
Step 5: Increment n, return to Step 2.

We will not discuss this further here, although the similarity of the wave system (8) and the recursion (Step 3) should convince the reader there is some connection. The asymmetric Schur algorithm may be used in place of the asymmetric Levinson algorithm; this is advantageous in a parallel computing environment.

G. Reconstruction of Medium from $\mathbf{r}_{n,n+1}$ and $\mathbf{s}_{n,n+1}$

A very important issue that arises when the reflection data are noisy is distinguishing actual interfaces from noise spikes in the reflection data that could be interpreted incorrectly as interfaces. Several authors, e.g., [15], have proposed thresholding the reflection data, so that a data value above the threshold is interpreted as an actual reflection, while a data value below threshold is regarded as noise and set to zero. This approach has proven quite promising for lossless media, and we propose to use it for lossy media as well (see Section VI).

The transfer coefficient $\mathbf{r}_{n,n+1}(\mathbf{s}_{n,n+1})$ for the real medium is the product (quotient) of the reflection coefficient $\{R_{n,n+1}\}$ at the boundary between the nth and $(n+1)$th layers and the cumulative loss factor $\{L_n = e^{-2\mathbf{A}_{n+1}}\}$ of the system up to this boundary. For the adjoint medium, $\mathbf{r}_{n,n+1}(\mathbf{s}_{n,n+1})$ is the quotient (product) of $R_{n,n+1}$ and L_n. For normal TE wave propagation $\{\theta_i = 0\}$ and $\{\alpha_{zi} = \alpha_i\}$, for all i,

$$\text{Lossless medium: } \mathbf{r}_{n,n+1} = R_{n,n+1}L_n = R_{n,n+1} = \frac{R_{n,n+1}}{L_n} = s_{n,n+1} \tag{10a}$$

$$\text{Lossy medium: } \mathbf{r}_{n,n+1} = R_{n,n+1}L_n,\ \mathbf{s}_{n,n+1} = \frac{R_{n,n+1}}{L_n}, \Rightarrow \mathbf{r}_{n,n+1}\mathbf{s}_{n,n+1} = R_{n,n+1}^2 \tag{10b}$$

where

$$R_{n,n+1} = \frac{\sqrt{\varepsilon_n} - \sqrt{\varepsilon_{n+1}}}{\sqrt{\varepsilon_n} - \sqrt{\varepsilon_{n+1}}} \tag{11a}$$

$$L_n = e^{-2\mathbf{A}_{n+1}} = e^{-2\sum_{i=0}^{n}\alpha_i\Delta_i}. \tag{11b}$$

The time interval $\{T_n\}$ is the two-way propagation time in the nth layer and it is given by the interval between the $\mathbf{r}_{n-1,n}$ and $\mathbf{r}_{n,n+1}$ transfer coefficients. From the transfer coefficients and time intervals, the medium can be completely reconstructed as follows:

IV. RECONSTRUCTION USING TWO ANGLES OF INCIDENCE

In [5], it has been shown that the impulse reflection and transmission responses of a lossy layered medium, from both sides of the medium, can be used to determine $r(n)$ and $\tilde{r}(n)$. However, in many applications (e.g., RES of a glacier) it is not practical to probe a medium from both sides. Thus, it behooves us to find a method for reconstructing a lossy medium purely from the reflection response on one side. Previous work [14] has shown that the one-sided reflection response is sufficient to reconstruct a *lossless* medium. In this section, we show that the one-sided reflection responses at *two angles* of incidence results in sufficient information to reconstruct a lossy medium. In practice, an impulsive point source would be used; such

a source can be decomposed into impulsive plane waves at various angles of incidence using the Weyl integral, and plane wave reflection responses at different angles can be separated by beam forming. The use of an impulsive plane wave at a specific angle $\{\theta_0\}$ allows us to reformulate 2-D problems as 1-D problem and use the formulation described in Section III.

A. Reconstruction of Permittivities from Time Intervals

We begin by assembling the normal incidence reflection response $\{r(n)\}$ into a *symmetric* Toeplitz system, i.e., let $\check{r}(n) = r(n)$ in (9). Running the *symmetric* Levinson algorithm on this system results in a single set of nonzero transfer coefficients $\{\hat{\mathbf{r}}_{i,i+1}\}$, which occur at time intervals $\{T_i\}$ (at other times the transfer coefficients are zero). Note that the Levinson recursions are necessary to suppress multiple reflections due to previous layers: this is called *predictive deconvolution*. Since the medium losses are not properly accounted for, these coefficients $\{\hat{\mathbf{r}}_{i,i+1}\}$ *do not have the same values as the true transfer coefficients* $\{\mathbf{r}_{i,i+1}, \mathbf{s}_{i,i+1}\}$. However, $\hat{\mathbf{r}}_{i,i+1}$, $\mathbf{r}_{i,i+1}$, and $\mathbf{s}_{i,i+1}$ will all occur at the *same times*, permitting recovery of the intervals $\{T_i\}$. If the medium has low losses, the Levinson recursions will reduce (but not eliminate, since $\hat{\mathbf{r}}_{i,i+1} \neq \mathbf{r}_{i,i+1}$) the multiple reflections enough to enable easy determination of the interfaces directly off the primary reflections.

Next, the medium is probed with an impulse at some nonzero angle $\{\theta_0\}$, resulting in the reflection response $\{r'(n)\}$. Assembling this response into a symmetric Toeplitz system (9), and running the symmetric Levinson algorithm, produces a different set of transfer coefficients $\{\hat{\mathbf{r}}'_{i,i+1}\}$. These transfer coefficients occur at time intervals $\{T'_i\}$. The time intervals for normal $\{T_i\}$ and oblique $\{T'_i\}$ sensing are proportional to the wavespeeds $\{v_i$ and v'_i, respectively$\}$ in each layer, and are related to the angle of travel $\{\theta_i\}$ (6) by

$$v'_i = \frac{\Delta_i}{T'_i} = v_{zi} = \frac{v_i}{\cos\theta_i} \rightarrow \theta_i = \cos^{-1}\frac{T'_i}{T_i}. \qquad (12)$$

Thus from the two sets of time intervals $\{T_i, T'_i\}$, the angle of propagation in each layer $\{\theta_i\}$ can be calculated. Using the low-loss assumption (6) $\{\beta_i \approx k_0\sqrt{\varepsilon'_i}\}$, Snell's law relates the angles of propagation in all layers by $\sqrt{\varepsilon'_i}\sin\theta_i = \sqrt{\varepsilon'_j}\sin\theta_j$. Since the angle of incidence $\{\theta_0\}$ and permittivity $\{\varepsilon_0\}$ of the medium from which probing takes place are known (e.g., for air $\varepsilon_0 = 1$), the permittivity of each layer $\{\varepsilon_i\}$ can be found recursively. From the permittivity $\{\varepsilon_i\}$ and travel-time $\{T_i\}$, the thickness of each layer is determined by $\Delta_i = \frac{T_i c}{2\sqrt{\varepsilon'_i}}$.

Once the permittivity of each layer is known, the reflection coefficient at each boundary can be calculated as $R_{i,i+1} = \frac{\sqrt{\varepsilon_i}-\sqrt{\varepsilon_{i+1}}}{\sqrt{\varepsilon_i}+\sqrt{\varepsilon_{i+1}}}$ for normal incidence and $R'_{i,i+1} = \frac{\sqrt{\varepsilon_i}\cos\theta_i - \sqrt{\varepsilon_{i+1}}\cos\theta_{i+1}}{\sqrt{\varepsilon_i}\cos\theta_i + \sqrt{\varepsilon_{i+1}}\cos\theta_{i+1}}$ for oblique incidence.

B. Reconstruction of Loss Factors from Reflection Response Values

Using reflection responses at more than two angles does not provide sufficient additional information to compute L_i and α_i. The reason for this is that L_i, like the travel-times T_i, do not affect the interface between layers. We were able to recover T_i in a layer by looking at the *interval* between $\hat{\mathbf{r}}_{i-1,i}$ and $\hat{\mathbf{r}}_{i,i+1}$ (note that the interval is equivalent to T_i, since the multiple reflections have been suppressed). However, determining L_i requires use of the *magnitude* of $\hat{\mathbf{r}}_{i,i+1}$, which cannot yet be interpreted.

We now introduce a novel method of reconstruction which involves running the Levinson algorithm *backward* to generate a synthetic response $\{\check{r}(n)\}$. We begin by simulating the response for a lossless version of the medium, i.e., $\check{\varepsilon}'_i = \varepsilon'_i$ but $\check{\varepsilon}''_i = 0$. This response is generated using the backward asymmetric Levinson algorithm with $\hat{\mathbf{r}}_{i,i+1} = \check{\mathbf{r}}_{i,i+1} = \check{\mathbf{s}}_{i,i+1}$. The backward asymmetric Levinson algorithm is as follows:

Backward Asymmetric Levinson Algorithm

Step 1: Initialize: $n = 0$, $A_0(z) = B_0(z) = 1$, and $\tau_0 = 1$.

Step 2:

$$\begin{bmatrix} A_{n+1}(z) \\ B_{n+1}(z) \end{bmatrix} = \frac{z^{+\frac{1}{2}}}{\sqrt{1-\check{\mathbf{r}}_{n,n+1}\check{\mathbf{s}}_{n,n+1}}} \begin{bmatrix} 1 & \check{\mathbf{s}}_{n,n+1} \\ \check{\mathbf{r}}_{n,n+1} & 1 \end{bmatrix} \times \begin{bmatrix} z^{-\frac{1}{2}} & 0 \\ 0 & z^{+\frac{1}{2}} \end{bmatrix} \begin{bmatrix} A_n(z) \\ B_n(z) \end{bmatrix}$$

Step 3: $\check{r}(n+1) = \check{\mathbf{r}}_{n,n+1}\tau_n + \sum_{i=1}^{n} a_{n,i}\check{r}(n+1-i)$

$$\check{\check{r}}(n+1) = \check{\mathbf{s}}_{n,n+1}\tau_n + \sum_{i=0}^{n-1} b_{n,i}\check{\check{r}}(i+1).$$

Step 4: $\tau_{n+1} = \tau_n\sqrt{1-\check{\mathbf{r}}_{n,n+1}\check{\mathbf{s}}_{n,n+1}}$

Step 5: Increment n, *return to Step* 2.

If the generated response $\{\check{r}(n)\}$ found with $\hat{\mathbf{r}}_{i,i+1} = \check{\mathbf{r}}_{i,i+1} = \check{\mathbf{s}}_{i,i+1}$ is equal to the measured response $\{r(n)\}$, then we have just verified the medium to be *lossless*. If on the other hand $\check{r}(n) \neq r(n)$, then we have verified the medium to be *lossy* and the values for $\check{\mathbf{r}}_{i,i+1}$ and $\check{\mathbf{s}}_{i,i+1}$ must be updated. When the medium has loss, the measured returns $\{r(n)\}$ are less in magnitude than those just generated $\{\check{r}(n)\}$, i.e., $|r(n)| \leq |\check{r}(n)|$. We note that the first nonzero return $\{r(n_0)\}$ is due to reflection off the first interface, and that its magnitude is determined solely by that boundary's *actual transfer* coefficients $\{\mathbf{r}_{0,1}\}$, whereas in our simulated medium, the first nonzero return $\{\check{r}(n_0)\}$ is determine by the first boundary's *reflection* coefficient $\{R_{0,1}\}$. *Recall that we know* $R_{0,1}$ *from* (3b), *since we reconstructed the permittivities from* ε_n *the interval information*. Since the transfer and reflection coefficients are related through the loss factor $\{L_0\}$, L_0 can be found using

$$L_0 = \frac{r(n_0)}{\check{r}(n_0)} \Rightarrow \mathbf{r}_{0,1} = R_{0,1}L_0 \quad \text{and} \quad \mathbf{s}_{0,1} = \frac{R_{0,1}}{L_0}. \qquad (13)$$

Note that we have now found the *exact* transfer coefficients for the first boundary. Since the loss in the first layer is reflected in the subsequent transfer coefficients, we can now generate trial values for the remaining lossy medium transfer coefficients $\{\check{\mathbf{r}}_{i,i+1}, \check{\mathbf{s}}_{i,i+1}\}$:

$$\check{\mathbf{r}}_{i,i+1} = R_{i,i+1}L_0 \quad \text{and} \quad \check{\mathbf{s}}_{i,i+1} = \frac{R_{i,i+1}}{L_0}, \forall i \in [0, t-1]. \qquad (14)$$

To find the losses in the subsequent layers, we generate two responses using the backward asymmetric Levinson algorithm. The first response $\{\check{r}(n)\}$ is our new synthetic medium impulse response generated by using the current trial values for the transfer coefficients for all layers $\{\check{\mathbf{r}}_{i,i+1}, \check{\mathbf{s}}_{i,i+1} \forall i \in [0, t-1]\}$. The second response $\{\dot{r}(n)\}$ is generated by using *only* the coefficients $\{\mathbf{r}_{i,i+1}, \mathbf{s}_{i,i+1} \forall i \in [0, p-1]\}$ that are known *exactly* after the first p layers have been identified; the remaining coefficients are set to zero. We once again compare the trial $\{\check{r}(n)\}$ with the measured $\{r(n)\}$ response. Any difference indicates incorrect values of the trial transfer coefficients.

Let $\check{r}(n_p)$ and $r(n_p)$ differ, indicating that our trial values for the transfer coefficients $\{\check{\mathbf{r}}_{p,p+1}, \check{\mathbf{s}}_{p,p+1}\}$ are incorrect. We correct the coefficients as follows:

$$L_p = L_{p-1}\frac{r(n_p) - \dot{r}(n_p)}{\check{r}(n_p) - \dot{r}(n_p)} \Rightarrow \mathbf{r}_{p,p+1} = R_{p,p+1}L_p \qquad (15)$$

andWe subtract $\dot{r}(n)$ from the measured $\{r(n)\}$ and trial $\{\check{r}(n)\}$ responses to remove the effects of multiple reflections up to and including time n_p due to the layers previously determined. A new layer has now been identified, and we update the trial values for the transfer coefficient for subsequent layers, using

$$\check{\mathbf{r}}_{i,i+1} = R_{i,i+1}L_p \quad \text{and} \quad \check{\mathbf{s}}_{i,i+1} = \frac{R_{i,i+1}}{L_p}, \forall i \in [p+1, t-1]. \qquad (16)$$

One other important point should be noted here. The first stage of our algorithm uses the *symmetric* Levinson algorithm to reduce multiple reflections and identify depths (in travel time) of interfaces. However, the symmetric Levinson algorithm incorrectly assumes a lossless medium. As a result, multiple reflections will not be eliminated, although if the medium losses are low they will be greatly suppressed, and if the medium losses are high they will be very small anyway. However, any significant multiple reflections may be misidentified as primary reflections from (nonexistent) interfaces.

If we know *a priori* that contiguous layers have significantly different properties, simply thresholding the reflection coefficients generated by the symmetric Levinson algorithm will distinguish the primary reflections from the multiple reflections. Otherwise we can check the consistency of the result of each recursion, again by comparing the trial $\{\check{r}(n)\}$ with the measured $\{r(n)\}$ responses. Any differences for $n < n_p$ indicates misidentification of an interface. This is another reason for this recursive use of layer stripping.

C. Summary of Algorithm

The reconstruction method for the discrete lossy dielectric is as follows:

Reconstruction of Lossy Dielectric Using Two Angles:

Step 1: Measure impulse reflection responses to normal $\{r(n)\}$ *and oblique* $\{r'(n)\}$ *probing.*

Step 2: Structure each response into a symmetric Toeplitz system (9); *solve this system using symmetric Levinson algorithm.*

Step 3: Determine time intervals $\{T_i, T_i', \}$ *between coefficients* $\{\hat{\mathbf{r}}, \hat{\mathbf{r}}'\}$.

Step 4: $\theta_i = \cos^{-1}\frac{T_i'}{T_i}, \forall i \in [1, t-1]$

Step 5: ε_0 *and* $\theta_i, \Rightarrow \sqrt{\varepsilon_i'} = \sqrt{\varepsilon_0'}\frac{\sin\theta_0}{\sin\theta_i}, \forall i \in [1, t-1]$.

Step 6: $R_{i,i+1} = \check{\mathbf{r}}_{i,i+1} = \check{\mathbf{s}}_{i,i+1} = \frac{\sqrt{\varepsilon_i} - \sqrt{\varepsilon_{i+1}}}{\sqrt{\varepsilon_i} + \sqrt{\varepsilon_{i+1}}}$.

Step 7: Initialze: $p = 0$, $L_{-1} = 1$, $L_{-1}' = 1$.

Step 8: Using backward asymmetric Levinson algorithm generate:

- $\dot{r}(n)$ *using transfer coefficients*

$$\mathbf{r}_{i,i+1}, \mathbf{s}_{i,i+1}, \text{ with } \mathbf{r}_{i,i+1} = \mathbf{s}_{i,i+1} = 0, \quad \forall i \in [p, t-1]$$

- $\check{r}(n)$ *using transfer coefficients*

$$\mathbf{r}_{i,i+1}, \mathbf{s}_{i,i+1}, \forall i \in [0, t-1]$$

Step 9:

$$L_p = L_{p-1}\frac{r(n_p) - \dot{r}(n_p)}{\check{r}(n_p) - \dot{r}(n_p)} L_p' = L_{p-1}' e^{+\ln\left(\frac{L_p}{L_{p-1}}\right)\cos\theta_p}$$

Step 10: $\mathbf{r}_{p,p+1} = R_{p,p+1}L_p$ *and* $\mathbf{s}_{p,p+1} = \frac{R_{p,p+1}}{L_p}$

$\check{\mathbf{r}}_{i,i+1} = R_{i,i+1}L_p$ *and* $\check{\mathbf{s}}_{i,i+1} = \frac{R_{i,i+1}}{L_p}, \quad \forall i \in [p+1, t-1]$

Step 11: Increment p, *return to Step* 8.

Step 12: With exact coefficients $\mathbf{r}_{i,i+1}, \mathbf{s}_{i,i+1}, \forall i \in [0, t-1]$, *implement lossy system reconstruction in Section III-G.*

D. Special Case: Low-Loss Medium

A low-loss medium is defined as one in which the attenuation constants in all layers and the reflection coefficients for all interfaces are both very small, i.e., $\alpha_i \ll 1$ and $R_{i,i+1} \ll 1$. Under these conditions, the following approximation is valid:

$$O\left(\sqrt{\varepsilon_i'}\right) \approx O\left(\sqrt{\varepsilon_j'}\right), \varepsilon_i' \gg \varepsilon_i'' \Rightarrow \hat{\mathbf{r}}_{i,i+1} \approx \mathbf{r}_{i,i+1} \qquad (17)$$
$$\text{and} \quad \hat{\mathbf{r}}_{i,i+1}' \approx \mathbf{r}_{i,i+1}', \forall i, j \in [0, t-1].$$

With this approximation the reconstruction of the medium is greatly simplified:

Reconstruction of Low-Loss Dielectric Medium

Step 1: Measure impulse reflection responses to normal $\{r(n)\}$ *and oblique* $\{r'(n)\}$ *probing.*

Step 2: Structure each response into a symmetric Toeplitz system (9); *solve this system using symmetric Levinson algorithm.*

Step 3: Determine time intervals $\{T_i, T_i', \}$ *between coefficients* $\{\hat{\mathbf{r}}, \hat{\mathbf{r}}'\}$.

Step 4: $\theta_i = \cos^{-1}\frac{T_i'}{T_i}, \forall i \in [1, t-1]$

Step 5: ε_0 *and* $\theta_i, \Rightarrow \sqrt{\varepsilon_i'} = \sqrt{\varepsilon_0'}\frac{\sin\theta_0}{\sin\theta_i}, \forall i \in [1, t-1]$.

Step 6: Initialize: $p = 0$, $L_{-1} = 1$, $L_{-1}' = 1$.

Step 7: $R_{i,i+1} = \check{\mathbf{r}}_{i,i+1} = \check{\mathbf{s}}_{i,i+1} = \frac{\sqrt{\varepsilon_i} - \sqrt{\varepsilon_{i+1}}}{\sqrt{\varepsilon_i} + \sqrt{\varepsilon_{i+1}}}, \forall i \in [0, t-1]$.

Step 8: $\Delta_i = 2T_i' \cdot \text{pole}$ of the ith $\in [0, t-1]$

Step 9: $L_i' = L_{i-1}' e^{+\ln\frac{1}{1-\Delta_i}\cos\theta_i}, \forall i \in [0, t-1]$.

Note, Steps 1–5 are the same as in Section IV-C, but repeated use of the backward asymmetric Levinson algorithm to generate synthetic reflection responses is no longer required.

V. Reconstruction of Unknown Bottom Layer

The methods presented so far are unable to reconstruct the dielectric constant of an unknown bottom half-space (the tth layer). This situation is often the case, e.g., RES probing of an ice shelf. Since the tth layer is semi-infinite, no returns are available from which to calculate the normal and oblique travel times $\{T_t, T'_t\}$. Without this information, we cannot calculate the angle of propagation $\{\theta_t\}$, the permittivity $\{\varepsilon'_t\}$, and the reflection coefficient $\{R_{t-1,t}\}$. Without the reflection coefficient, the transfer coefficients $\{\mathbf{r}_{t-1,t}, \mathbf{s}_{t-1,t}\}$, from which the attenuation coefficient of the $(t-1)$th layer $\{\alpha_{t-1}\}$ is obtained, cannot be calculated.

We now show that by using two-angle probing, we have sufficient information for complete reconstruction of the layered medium, including the bottom half-space. The desired quantities $\{\varepsilon'_t$ and $\alpha_{t-1}\}$ may be found from the normal and oblique reflection responses $\{r(n), r'(n)\}$ as follows. From the procedure given in Section IV, the following are known:

- permittivity of the $(t-1)$th-layer $\{\varepsilon'_{t-1}\}$
- angle of propagation in $(t-1)$th-layer $\{\theta_{t-1}\}$
- thickness of the $(t-1)$th-layer $\{\Delta_{t-1}\}$
- loss factors to the $t-2$, $t-1$ interface $\{L_{t-2}, L'_{t-2}\}$
- transfer and transmission coefficients $\{\mathbf{r}_{i,i+1}, \mathbf{s}_{i,i+1}, \tau_{i+1}, \forall i \in [0, t-2]\}$.

Since we have no information regarding the tth layer, we begin by setting the last reflection coefficient to zero, i.e., $R_{t-1,t} = 0$. Then, using the known set of transfer coefficients $\{\mathbf{r}_{i,i+1}, \mathbf{s}_{i,i+1}, \forall i \in [0, t-2]\}$, the synthesized normal response $\{\check{r}(n)\}$ will equal the measured response $\{r(n)\}$ up to time increment n_{t-1}. At n_{t-1}, the measured response contains reflection data from the last interface $\{t-1, t\}$, while the synthesized responses do not.

Using the Levinson algorithm, transfer coefficient for the next layer $\{\mathbf{r}_{t-1,t}\}$ can be found from the measured $\{r(n_{t-1})\}$ and estimated $\{\check{r}(n_{t-1})\}$ normal responses as

$$\mathbf{r}_{t-1,t} = -\frac{a_{n_{t-1},0}}{\tau_{n_{t-1}}}(\check{r}(n_{t-1}) - r(n_{t-1})). \tag{18}$$

Use the algorithm of Section IV to determine the oblique transfer coefficients $\{\mathbf{r}'_{i,i+1}, \mathbf{s}'_{i,i+1}, \forall i \in [0, t-2]\}$. Using these coefficients and appropriate time delays $\{t'_i, \forall i \in [0, t-1]\}$ we generate an estimate $\{\check{r}'(m)\}$ of the oblique response. As in (18), the oblique transfer coefficient $\{\mathbf{r}'_{t-1,t}\}$ for the last layer is found from the measured $\{r'(m_{t-1})\}$ and estimated $\{\check{r}'(m_{t-1})\}$ responses.

Note that the time indices n_{t-1} for the normal probing responses $\{r(n_{t-1}), \check{r}(n_{t-1})\}$, and m_{t-1} for the oblique probing responses $\{r(m_{t-1}), \check{r}'(m_{t-1})\}$, are not the same. However, they are indicative of the same interface, i.e., the boundary between the $(t-1)$th and tth layer. Hence

$$\mathbf{r}'_{t-1,t} = -\frac{a'_{m_{t-1},0}}{\tau'_{m_{t-1}}}(\check{r}'(m_{t-1}) - r'(m_{t-1})). \tag{19}$$

Recall from (3) and (15) that the transfer coefficients for normal $\{\mathbf{r}_{t-1,t}\}$ and oblique $\{\mathbf{r}'_{t-1,t}\}$ probing are given by

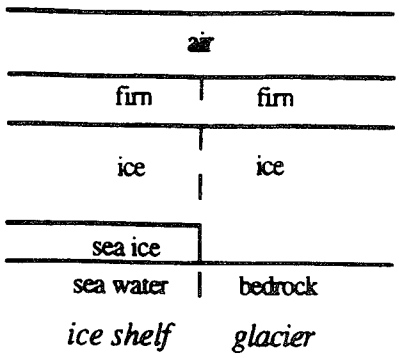


Fig. 4. Ice shelf model.

$$\mathbf{r}_{t-1,t} = R_{t-1,t}L_{t-1} \Rightarrow |\mathbf{r}_{t-1,t}| = \left|\frac{\sqrt{\varepsilon_{t-1}} - \sqrt{\varepsilon_t}}{\sqrt{\varepsilon_{t-1}} + \sqrt{\varepsilon_t}}\right| e^{-2\alpha_{t-1}\Delta_{t-1}} L_{t-2}; \tag{20a}$$

$$\mathbf{r}'_{t-1,t} = R'_{t-1,t}L'_{t-1} \Rightarrow |\mathbf{r}'_{t-1,t}| = \left|\frac{\sqrt{\varepsilon_{t-1}}\cos\theta_{t-1} - \sqrt{\varepsilon_t}\cos\theta_t}{\sqrt{\varepsilon_{t-1}}\cos\theta_{t-1} + \sqrt{\varepsilon_t}\cos\theta_t}\right| \times e^{-2\alpha_{t-1}\Delta_{t-1}\cos\theta_{t-1}} L'_{t-2}. \tag{20b}$$

We now have two equations with three unknowns $\{\sqrt{\varepsilon_t}, \theta_t$ and $\alpha_{t-1}\}$. From Snell's law, the angle of travel in the bottom layer can be found from its permittivity, i.e., $\theta_t = \sin^{-1}\left(\frac{\sqrt{\varepsilon_{t-1}}}{\sqrt{\varepsilon_t}}\sin\theta_{t-1}\right)$. Using this in (20b) results in two equations in two unknowns $\{\alpha_{t-1}, \sqrt{\varepsilon_t}\}$. Solving for the attenuation coefficient $\{\alpha_{t-1}\}$ as a function of permittivity $\{\sqrt{\varepsilon_t}\}$ gives the following parametric equations:

$$\alpha_{t-1} = -\frac{1}{2\Delta_{t-1}}\ln\left(\frac{\mathbf{r}_{t-1,t}}{L_{t-2}} \times \frac{\sqrt{\varepsilon_{t-1}} + \sqrt{\varepsilon_t}}{\sqrt{\varepsilon_{t-1}} - \sqrt{\varepsilon_t}}\right); \tag{21a}$$

$$\alpha'_{t-1} = -\frac{1}{2\Delta_{t-1}\cos\theta_{t-1}}\ln\left(\frac{\mathbf{r}'_{t-1,t}}{L'_{t-2}} \times \frac{\sqrt{\varepsilon_{t-1}}\cos\theta_{t-1} + \sqrt{\varepsilon_t}\cos\left(\sin^{-1}\left(\frac{\sqrt{\varepsilon_{t-1}}}{\sqrt{\varepsilon_t}}\sin\theta_{t-1}\right)\right)}{\sqrt{\varepsilon_{t-1}}\cos\theta_{t-1} - \sqrt{\varepsilon_t}\cos\left(\sin^{-1}\left(\frac{\sqrt{\varepsilon_{t-1}}}{\sqrt{\varepsilon_t}}\sin\theta_{t-1}\right)\right)}\right). \tag{21b}$$

The solution to (21) is that value of ε_t for which $\alpha_{t-1} = \alpha'_{t-1}$.

VI. Numerical Example

A. Ice Shelf Model

To demonstrate the above algorithms, we simulate RES probing of an ice shelf, which is a floating glacial mass attached to land (Fig. 4). In addition to a shallow layer of snow/firn on top, there is a layer of salt ice at the interface between the glacial ice and the sea water. Ice shelves represent a stratified system that is of particular interest in glaciology, especially radioglaciology, and in physical oceanography [8]. Since layers of firn and sea ice will give poor results for ice thickness if the multiple reflections are neglected, a multi-layer model must be used (see Fig. 1).

TABLE II
ICE SHELF PARAMETERS (NORMAL PROBING)

Layer	Thickness (m)	Temp (°C)	Density ($\rho = \frac{g}{m^3}$)	Salinity ($S = ‰$)	ε'	2-way delay (nsec)	ε''	α	$e^{-2\alpha\Delta}$
Air	17.3	n/a	n/a	n/a	1	115.5	0	0	1
Firn	22.1	-25	.55	n/a	2.09	213.0	1.26e-4	9.1e-5	.9960
Ice	91.3	-15	n/a	n/a	2.95	1045.0	.0011	6.7e-4	.8848
Sea Ice	16.8	-10	n/a	5	3.44	207.5	.0382	2.2e-2	.4775
Sea Water	semi-∞	-1.5	n/a	10	84.41	∞	157	n/a	0

TABLE III
ICE SHELF PARAMETERS (OBLIQUE PROBING)

Layer	Angle (θ)	Distance (m)	t_n (nsec)	$e^{-2\alpha\Delta}$
Air	30.00°	15.0	100.0	1.000
Firn	20.23°	20.8	200.0	.9962
Ice	16.92°	87.3	1000.0	.8896
Sea Ice	15.63°	16.2	200.0	.4909
Sea Water	3.11°	semi-∞	∞	0

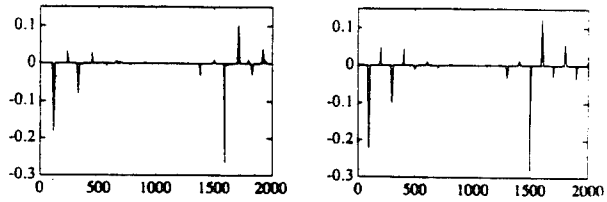


Fig. 5. Reflection response versus. nsec.

Furthermore, the layer of salt ice (or sea ice) is highly absorbing, compared to the firn and ice layers. We show that if the system is modeled as a lossless medium, the bottom is mistaken for bedrock $\{\varepsilon' \approx 10.5\}$ instead of sea water $\{\varepsilon' = 84.16\}$. On the other hand, if a low-loss model is assumed $\{\varepsilon' \gg \varepsilon''\}$, and we probe at two angles, the ice shelf is correctly reconstructed.

As shown in Fig. 4, the ice shelf is modeled as a multi-layer system. The thicknesses of the layers are realistic [7]–[9], [11], [12], [16]–[18], but they have been chosen to provide travel-times that are integer multiples of 0.5 nsec (this could be accomplished by integrating the continuous-time radar return and sampling every 0.5 nsec). We consider a video pulse system operating at 100 MHz. The transmitted pulses are 0.5 nsec in duration, and returns are clocked at 2 GHz. The dielectric constant $\{\varepsilon = \varepsilon' - i\varepsilon''\}$ for each layer was determined using a combination of actual measurements [8], [9] and theoretical values calculated using variations of the Debye equation [19]. Layer parameters for normal and oblique (30°) probing are given in Tables II and III, respectively.

The reflection and transfer coefficients for each interface are shown in Table IV.

B. Data: Plane-Wave Impulse Reflection Responses

Using the transfer coefficients given in Table IV, and the backward asymmetric Levinson algorithm, we simulated the free-surface reflection responses for normal $\{r(n)\}$ and oblique $\{r'(n)\}$ probing. Results are shown in Fig. 5(a) and (b), respectively.

As expected, the oblique response of an impulsive *plane wave* of infinite extent shows less travel delay due to the path length being shorter by a factor of $\cos\theta$. Also note the multiple reflections due to multiple scattering between the interfaces. Elimination of the multiple reflections using the wave system (8) that accounts for them is clearly necessary. Note that methods based on the Born (single-scattering) approximation will incorrectly identify these as (fictitious) interfaces. The effects of these errors will be exacerbated due to the losses in the system. A multiple reflection incorrectly identified as a primary reflection in a Born-approximation-based inversion method will have an incorrect loss factor assigned to the layer, and this introduces more error in reconstructing deeper layers.

C. Reconstruction of Reflection Coefficients

Solving the symmetric Toeplitz system constructed using the normal response $\{r(n)\}$ gives the reflection coefficients $\{\hat{r}_{n,n+1}\}$. Similarly, using the oblique response $\{r'(n)\}$ gives the oblique reflection coefficients $\{\hat{r}'_{n,n+1}\}$. The reflection coefficients are given in Table V.

D. Reconstruction Using Lossless Model

The medium reconstructed under the incorrect assumption that the medium is lossless, using the lossless algorithm in Section III-G and the data in Fig. 5(a), is given below in Table VI. We initialize the algorithm using the *a priori* knowledge of 0th layer. Note that ε' is off by approximately 2% in layer 3 and completely wrong in layer 4. This leads to misidentification of layer 3 as fractured ice instead of sea ice, and layer 4 as bedrock instead of sea water!

E. Reconstruction Using Lossy Model

We now use oblique plane-wave probing, performed at an angle of 30° relative to normal. Recall that this can be

TABLE IV
MODEL TRANSFER COEFFICIENTS

Interface $(n, n+1)$	$R_{n,n+1}$	L_n	$\mathbf{r}_{n,n+1}$	$\mathbf{s}_{n,n+1}$	$R'_{n,n+1}$	L'_n	$\mathbf{r}'_{n,n+1}$	$\mathbf{s}'_{n,n+1}$
Air/Firn (0,1)	-.1822	1.000	-.1822	-.1822	-.2207	1.000	-.2207	-.2207
Firn/Ice (1,2)	-.0859	.9960	-.0856	-.0862	-.0956	.9962	-.0952	-.0960
Ice/Sea Ice (2,3)	-.0384	.8813	-.0338	-.0436	-.0417	.8862	-.0370	-.0471
Sea Ice/Sea Water (3,4)	-.6641	.4208	-.2795	-1.5782	-.6741	.4350	-.2932	-1.5497

TABLE V
RECONSTRUCTED REFLECTION COEFFICIENTS

Interface $(n, n+1)$	$\hat{\mathbf{r}}_{n,n+1}$	$\hat{\mathbf{r}}'_{n,n+1}$
Air/Layer 1	-.1822	-.2207
Layer 1/Layer 2	-.0856	-.0952
Layer 2/Layer 3	-.0338	-.0370
Layer 3/Layer 4	-.2794	-.2931

TABLE VI
MEDIUM RECONSTRUCTION ASSUMING LOSSLESS MODEL

Layer (n)	ε'	t_n (nsec)	Thickness (m)	Type
0	1.00	115.5	17.3	air
1	2.09	213.0	22.1	firn
2	2.95	1045.0	91.3	glacial ice
3	3.38	207.5	16.9	fractured ice
4	10.65	∞	∞	bedrock w/ water!

accomplished using impulsive point source reflection data by beamforming for both normal and oblique incidence.

The angle of travel in each layer can be found from the ratio of normal to oblique travel intervals. The permittivity of each layer is then found using Snell's law, and the layer thickness can then be found knowing the travel time, angle, and permittivity. Results are given in Table VII-A. Note that the permittivities are not exact since we have limited the time interval accuracy to 0.5 nsec; choosing a smaller time interval would improve this. We have deliberately chosen a relatively large time interval to show that our method still works reasonably well.

From Tables IV and V, we note that the recovered reflection coefficients for both normal and oblique probing are approximately equal to the real transfer coefficients, i.e., $\hat{\mathbf{r}}_{n,n+1} \approx \mathbf{r}_{n,n+1}$ and $\hat{\mathbf{r}}'_{n,n+1} \approx \mathbf{r}'_{n,n+1}$. Thus we may use the low-loss assumption to reconstruct the layers (in actual application this would have to be known *a priori*; otherwise the full algorithm of Section IV-C must be used). From the calculated permittivities of Table VII–A and the measured coefficients $\{\hat{\mathbf{r}}'_{n,n+1}\}$ of Table V we can calculate the loss factor $\{L_n\}$ for all but the next to last layer using the low-loss algorithm of Section IV-D. Results are in Table VII-B.

TABLE VII-A
LAYER PARAMETERS FROM TWO-ANGLE PROBING

Layer (n)	t_n (nsec)	t'_n (nsec)	θ_n	ϵ_n	Δ_n (m)
0	115.5	100.0	30.0	1.00	17.3
1	213.0	200.0	20.1	2.10	22.0
2	1045.0	1000.0	16.9	2.95	91.1
3	207.5	200.0	**15.5**	**3.50**	**16.6**
4	∞	∞	?	?	∞

TABLE VII-B
CALCULATED LAYER LOSSES

Interface $(n, n+1)$	$\hat{\mathbf{r}}_{n,n+1}$	$R_{n,n+1}$	L_n	$\hat{\mathbf{r}}'_{n,n+1}$	L'_n
Air/Layer 1	-0.1822	-0.1822	1.000	-0.2207	1.000
Layer 1/Layer 2	-0.0856	-0.0859	0.9965	-0.0952	0.9967
Layer 2/Layer 3	-0.0338	-0.0427	**0.7916**	-0.0370	**0.7997**
Layer 3/Layer 4	**-0.2794**	-0.6540	-0.4272	**-0.2931**	0.4414

TABLE VIII
RECONSTRUCTION WITH TWO-ANGLE DATA

Layer n	ε_n	Δ_n (m)	α_n	Type
0	1.00	17.3	0	air
1	2.10	22.0	8.0e-5	firn
2	2.95	91.1	1.3e-3	ice
3	3.50	16.6	1.9e-2	sea ice
4	>80	∞	n/a	sea water

Using the boldface values in Tables VII–A and VII–B and (21a) and (21b), the remaining unknowns $\{\alpha_3$ and $\varepsilon_4\}$ can be found. Since reconstruction accuracy is limited by the 0.5 nsec sampling rate, minimizing the objective function $|\alpha_3 - \alpha'_3|$ does not produce a sharp null for any single ε_4. We do however find that the solution of must lie in the region $\varepsilon_4 > 80$. Therefore, we conclude the bottom half-space is sea water. Using $\varepsilon_4 \approx 80$, we calculate the values for α_3 and $R_{3,4}$ shown in italics in Table VII-B.

The final reconstructed values of the ice shelf are given in Table VIII and should be compared to the actual values in Table II. Here again, the knowledge of the 0th layer was *a priori*. We note that the final layer is now correctly identified as sea water, as opposed to bedrock in the lossless reconstruction. The permittivities ε' for the first three layers are reconstructed exactly; the error for the fourth layer is approximately 2%.

TABLE IX
RECONSTRUCTED REFLECTION COEFFICIENTS FOR LOW NOISE DATA

time (nsec)	130	250	340	1390	1600	1720	1840	1930
actual	-0.1822	0	-0.0856	-0.0338	-0.2795	0	0	0
Levinson	-0.1834	0	-0.0881	-0.0303	-0.2857	0	0	0
Born	-0.1873	0.0360	-0.0845	-0.0282	-0.2706	0.1012	-0.0260	0.0327

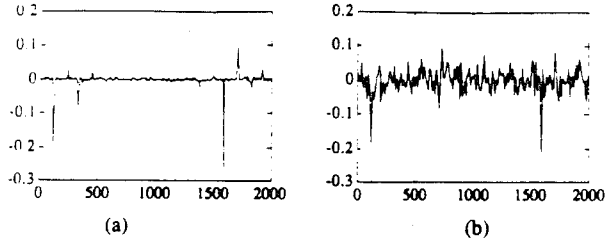


Fig. 6. Noisy versions of normal probing returns vs. nsec.

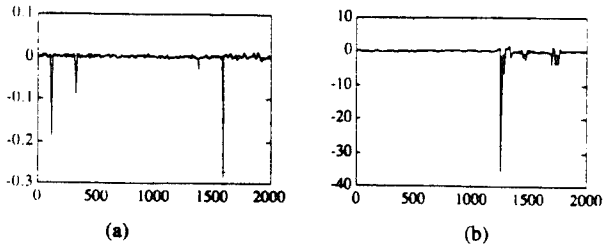


Fig. 7. Reconstructed transfer coefficients using unmodified reflection data.

Note that the algorithm computes the cumulative losses $\{L_n\}$ from which the attenuation coefficients $\{\alpha_n\}$ are subsequently computed. Round-off error can lead to an incorrect α_n, but the errors will not accumulate; note that α_3 is computed more accurately than α_2. The errors in permittivity and attenuation constants are due entirely to limiting the sample rate to 2 GHz; a finer discretization will produce better results. This example thus shows that proper modeling significantly improves the reconstruction results.

F. Reconstruction With Noisy Data

The layer stripping algorithms used in this reconstruction have the reputation of being unstable in noise. However, these algorithms fail only when given infeasible data, i.e., data that could not have come from an actual medium in a noise free environment. The condition for feasible *lossless* data is that the *symmetric* Toeplitz matrix in (9) be positive definite. For noisy data, the Toeplitz matrix may not meet this criteria, and this will cause the algorithms to diverge. In these situations, the Toeplitz matrix can be forced to be positive definite by setting all negative eigenvalues to some small positive number.

To illustrate this point we added zero-mean white Gaussian noise to the normal probing response discussed in the above example. Since this model was shown to be low-loss (i.e., $\hat{\mathbf{r}}_{i,i+1} \approx \mathbf{r}_{i,i+1}$), we argue that the response $\{r(n)\}$ must come close to meeting the lossless feasibility condition; therefore, we ensure it does by altering its eigenvalues as discussed (this is the projection of the Toeplitz matrix onto the space of positive definite matrices). We added both low (standard deviation = 0.003) and high (standard deviation = 0.03) levels of Gaussian noise to the normal response $\{r(n)\}$; the results are shown in Fig. 6(a) and (b), respectively.

Without implementing the data feasibility constraint, we applied the symmetric Levinson algorithm to the noisy reflection data $\{r(n)\}$. This gave the transfer coefficients $\{\hat{\mathbf{r}}\}$ shown in Fig. 7.

The reconstruction using the high noise data is shown in Fig. 7(b). Note the impossibly high $\hat{\mathbf{r}}_{i,i+1} \approx -35$; this marks divergence of the algorithm. In fact, the noise level was the *minimum* needed to cause the routine to fail. In contrast, the low noise reconstruction shown in Fig. 7(a) is merely a noisy version of the true reflection coefficient profile.

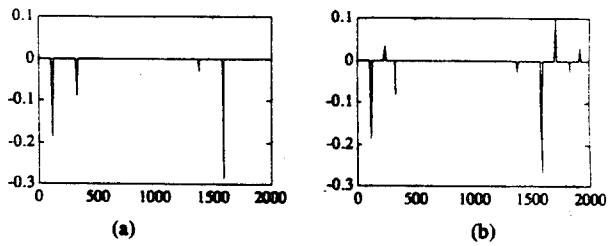


Fig. 8. Thresholded reflection data for low noise case. a) Levisons reconstruction, b) Born reconstruction.

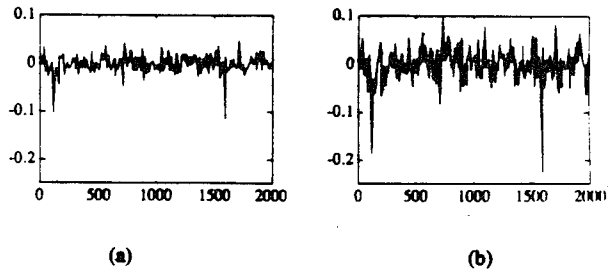


Fig. 9. Reflection coefficient reconstruction of high noise data. (a) Levinson reconstruction, (b) Born reconstruction.

As proposed earlier, we were able to improve this result by thresholding. Any estimated $\hat{\mathbf{r}}_{i,i+1}$ less than 0.025 in magnitude was set to zero. We also implemented a Born approximation (i.e., neglect multiple reflections) on the low noise data and applied the same threshold. The results are shown in Fig. 8(a) and (b).

We note that the Levinson reconstruction is superior to the Born approximation in that no false interfaces were obtained and that the reconstructed reflection coefficients are closer to the true values (Table IX).

We now investigate reconstruction with high noise data. Since the original reconstruction diverged, we applied the positive definite constraint to the symmetric Toeplitz response matrix. The symmetric Levinson algorithm was then run on the modified Toeplitz response matrix, with its result plotted in Fig. 9(a). Again we compare this reconstruction with that obtained using the Born approximation [Fig. 9(b)].

In the presence of this much noise it is hard to argue that one routine is superior to another, (although it does appear that the Born approximation gives better estimates of the coefficients).

The main conclusion of all this is that the stability of the Levinson reconstruction is data dependent and when the *symmetric* Levinson algorithm is used for the inverse problem, the data can be constrained such that stability is ensured. Another possibility is to use plane-wave reflection data at more than two angles of incidence. Although the additional angles provide no extra information for noiseless data, they can be used to compute least-squares estimates of the layer parameters. This was done in [4] for the acoustic medium, and significant improvement was observed.

VII. Conclusion

We have presented and discussed some new algorithms for reconstructing a lossy layered dielectric from its reflection responses to plane waves at two different angles of incidence. Losses are modeled by an asymmetric wave system associated with asymmetric Toeplitz matrices and the asymmetric Levinson algorithm. This permits rapid solution of the forward and inverse scattering problems. A novel approach that iterates between these two problems, recursively reconstructing another layer of the medium at each iteration, was also presented; this approach is needed since absorption is a property of the layers, not the interfaces, so probing at additional angles will not help. Numerical simulations of an ice shelf demonstrated that the algorithms work, and provide significant improvement over models that neglect losses (lossless model) and multiple reflections (the Born approximation). Performance of the reconstruction algorithms using noisy response data was discussed.

A companion paper [5] has discussed the asymmetric wave system (8) in greater detail, along with algorithms for forward and inverse scattering that involve transmission as well as reflection responses. In the present paper we have focused on reconstruction of lossy media from reflection responses only, since transmission responses are generally unavailable in remote sensing applications. Further applications in remote sensing, noted in Section II, and nondestructive testing (for which transmission data can be used), constitute possible topics for further research.

References

[1] T. M. Habashy and R. Mittra, "On some inverse methods in electromagnetics," *J. Electromagnetic Waves Appl.*, vol. 1, no. 1, pp. 25–58, 1987.

[2] T. Uno and S. Adachi, "Inverse scattering method for one-dimensional inhomogeneous layered media," *IEEE Trans. Antenna Propagat.*, vol. AP-35, no. 12, Dec. 1987.

[3] E. A. Hashish, M. M. Zahran, and M. N. I. Fahmy, "New method for permittivity and conductivity inversion of discrete layered earth," *Proc. IEE*, vol. 137, Pt. A, pp. 346–348, 1980.

[4] A. E. Yagle, "Numerical performance of layer stripping algorithms for a layered medium probed at nonnormal incidence," in M. Worthington, Ed. *Deconvolution and Inversion*. Oxford, UK: Blackwell, 1987.

[5] J. Frolik and A. E. Yagle, "Forward and inverse scattering for discrete layered lossy and absorbing media," submitted to *SIAM J. Applied Math.*, June 1993.

[6] F. Ulaby and C. Elachi, *Radar Polarimetry for Geoscience Applications*. Norwood, MA: Artech House, 1990.

[7] J. Bamber and J. Dowdeswell, "Remote Sensing Studies of Kvitoyjokulen," *J. Glaciology*, vol. 36, no. 122, 1990.

[8] V. Bogorodsky, C. Bentley, and P. Gudmandsen, *Radioglaciology*. Dordrect, Netherlands: D. Reidel, 1985.

[9] D. Hall and J. Martinec, *Remote Sensing of Ice and Snow*. London: Chapman and Hall, 1985.

[10] R. Goodman, "Radio echo sounding on temperate glaciers," *J. Glaciology*, vol. 15, no. 73, pp. 57–69, 1975.

[11] J. Dowdeswell *et al.*, *Airborne Radio Echo Sounding of Sub-Polar Glaciers in Spitsbergen*. Norsk Polarinstitutt, 1984.

[12] D. Wright *et. al.*, "A digital low-frequency, surface profiling ice-radar system," *J. Glaciology*, vol. 36, no. 122, pp. 112–121, 1990.

[13] R. Harrington, *Time-Harmonic Electromagnetic Fields*. New York: McGraw-Hill, 1961.

[14] A. M. Bruckstein and T. Kailath, "Inverse scattering for discrete transmission-line models," *SIAM Rev.*, vol. 29, no. 3, Sept. 1987

[15] A. M. Bruckstein, I. Koltracht, and T. Kailath, "Inverse scattering with noisy data," *SIAM J. Sci. Stat. Comput.*, vol. 7, no. 4, pp. 1331–49, Oct. 1986.

[16] W. Paterson, *Physics of Glaciers*. New York: Pergamon Press, 1969.

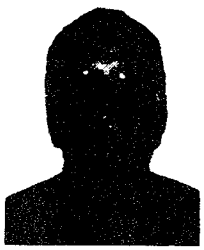
[17] A. Annan and J. Davis, "Impulse radar sounding in permafrost," *Radio Sci.*, vol. 11, pp. 383–94, 1976.

[18] G. Raju, W. Xin, and R. Moore, "Design, development, field observations, and preliminary results of the coherent Antarctic radar depth sounder (CARDS)," *J. Glaciology*, vol. 36, no. 123, pp. 247–254, 1990.

[19] F. Ulaby, R. Moore, and A. Fung, *Microwave Remote Sensing*, vol. I-III. Norwood, MA: Artech House, 1981.

[20] T. M. Habashy, W. C. Chew, and E. Y. Chow, "Simultaneous reconstruction of permittivity and conductivity profiles in a radially inhomogeneous slab," *Radio Sci.*, vol. 21, no. 4, pp. 635–45, July 1986.

[21] D. Lesselier, "Optimization techniques and inverse problems: reconstruction of conductivity profiles in the time domain," *IEEE Trans. Antenna Propagat.*, vol. AP-30, no. 1, pp. 55–65, 1982.



Jeff Frolik (S'83) received the B.S.E.E. degree from the University of South Alabama, Mobile, and the M.S.E.E. degree from the University of Southern California, Los Angeles, in 1986 and 1988, respectively. He is pursuing the Ph.D. degree in electrical engineering at the University of Michigan, Ann Arbor.

He has been an employee of the Hughes Aircraft Company since 1986. His current research interests are in inverse problems, fast algorithms, and signal detection and classification.

Andrew Yagle (M'85) was born in Ann Arbor, MI, in 1956. He received the B.S.E. and B.S.E.E. degrees from the University of Michigan, Ann Arbor and the S.M., E.E., and Ph.D. degrees from the Massachusetts Institute of Technology, Cambridge, in 1977, 1978, 1981, 1982, and 1985, respectively.

Since September 1985, he has been with the Department of Electrical Engineering and Computer Science, University of Michigan, where he is currently an Associate Professor. His research interests include fast algorithms for digital signal processing, multiresolution and iterative algorithms in medical imaging, multidimensional inverse scattering, phase retrieval, and linear least-squares estimation.

Dr. Yagle received the National Science Foundation Young Investigator Award in 1988 and the ONR Young Investigator Award in 1990. He received the H.H. Rackham School of Graduate Studies Research Partnership Awards in 1990 and 1993. He received the College of Engineering Teaching Excellence Award in 1992, the Eta Kappa Nu Professor of the Year Award in 1990, and the Class of 1938e Distinguished Service Award in 1989. He is currently an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE SIGNAL PROCESSING LETTERS and Multidimensional Systems and Signal Processing. He is also a member of the Digital Signal Processing Technical Committee, a past Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING, and cotechnical chair of ICASSP-95, to be held in Detroit, MI.

APPENDIX F1

T.-S. Pan and A.E. Yagle, "Acceleration and Filtering in the Generalized Landweber Iteration using a Variable Shaping Matrix," IEEE Trans. Medical Imaging 12(2), 278-286, June 1993.

This paper discusses the generalized Landweber iteration for solving large linear systems of equations. We show how the convergence and filtering behavior of this algorithm can be tightly controlled, in contrast to most iterative algorithms whose behavior cannot be controlled and that simply go where they may. This paper is a good introduction to the algorithm and how to design it to obtain desired behavior.

Although the specific application investigated here is positron emission tomography, the results could also be applied to discretized integral equations, such as the generalized Gel'fand-Levitan or Marchenko integral equations. Although the matrix kernels are no longer sparse, a projection or backprojection (multiplication by the kernel or its transpose) can be implemented quickly using FFT-based convolution methods; number-theoretic transforms would require even fewer multiplications.

Acceleration and Filtering in the Generalized Landweber Iteration Using a Variable Shaping Matrix

Tin-Su Pan, *Member, IEEE*, Andrew E. Yagle, *Member, IEEE*, Neal H. Clinthorne, *Member, IEEE*, and W. Leslie Rogers, *Member, IEEE*

Abstract— **We use the generalized Landweber iteration with a variable shaping matrix to solve the large linear system of equations arising in the image reconstruction problem of emission tomography. Our method is based on the property that once a spatial frequency image component is almost recovered within ϵ in the generalized Landweber iteration, this component will still stay within ϵ during subsequent iterations with a different shaping matrix, as long as this shaping matrix satisfies the convergence criterion for the component. Two different shaping matrices are used: the first recovers low-frequency image components; and the second may be used either to accelerate the reconstruction of high-frequency image components, or to attenuate these components to filter the image. The variable shaping matrix gives results similar to truncated inverse filtering, but requires much less computation and memory, since it does not rely on the singular value decomposition.**

I. Introduction

In emission tomography, the image reconstruction problem can be formulated as the solution to a large linear system of equations [1]–[3]. This system of equations tends to be large, with dimensions on the order of thousands, and ill-conditioned. For this reason, regularization and minimization of computation time in solving it become important issues.

One solution [4]–[7] is to perform a singular value decomposition (SVD) of the system matrix which describes the transformation or projection process of the imaging system. Since the system matrix is usually ill-conditioned, small singular values can be set to zero to obtain a stable (not sensitive to the measurement noise) solution. This approach is called truncated inverse filtering (TIF) [8]. Another approach is to window the small singular values to zero through a decreasing series of weight factors while keeping the large singular values unchanged. This research follows the latter approach.

Manuscript received November 13, 1991; revised July 10, 1992. The work of T.-S. Pan was supported in part by NIH Grant PO1-CA42768, and in part by a Research Partnership award from the H. H. Rackham School of Graduate Studies of the University of Michigan. The work of A. E. Yagle, N. H. Clinthorne, and W. L. Rogers was supported in part by ONR Grant N00014-90-J-1897, NIH Grant RO1-CA32846, and NIH Grant RO1-CA54362, respectively.

T.-S. Pan was with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI. He is now with the Department of Nuclear Medicine of the University of Massachusetts Medical Center, 55 Lake Ave. North, Worcester, MA 01655.

A. E. Yagle is with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109-2122.

N. H. Clinthorne and W. L. Rogers are with the Division of Nuclear Medicine, University of Michigan Medical Center, Ann Arbor, MI 48109-0552.

IEEE Log Number 9208170.

Due to the attenuation, the imaging system in emission tomography is spatially variant. The point spread function of the imaging system varies with the location of the point source. Fourier spatial frequencies are unsuitable for describing orthogonal components of the image. Instead, this paper defines "spatial frequencies" as the reciprocal of singular values, and the associated image components as the projection of the image on the corresponding singular vectors. This terminology follows [6], [7]. Therefore, high spatial frequency components of the object are defined as components on singular vectors associated with small singular values; and low spatial frequency components as components on singular vectors with large singular values [9], [10]. Using this terminology, high-frequency components of the reconstructed image, which are sensitive to noise, will be removed in TIF.

To compute the SVD of a large system matrix will take a long time and much memory space. Furthermore, in emission tomography, attenuation results in different objects having different system matrices, each of which would require an SVD. Thus, using TIF to derive a solution for the image reconstruction in emission tomography requires a huge amount of computation and memory space.

This paper presents a novel and alternative approach—using the generalized Landweber iteration [11] with a variable shaping matrix (defined below). This method can either 1) accelerate the reconstruction of high-frequency components; or 2) roll off the inverse filter to prevent Gibbs phenomenon arising from the sharp truncation in TIF, and preserve the stability of the solution when the system matrix is ill-conditioned. The motivation for using a variable shaping matrix is based on the recover-and-stay property of the generalized Landweber iteration which is stated and proved for the first time in this paper. This property states that *in the generalized Landweber iteration, once an image component is recovered within ϵ, the component will still stay within ϵ in subsequent iterations even with a different shaping matrix, as long as this shaping matrix satisfies the convergence criterion for the component.* Our new approach uses two different shaping matrices: the first one is for fast recovery of low-frequency components and partial recovery of high-frequency components; the second one may be used either for speeding up the reconstruction of high-frequency components (if noise is not a problem), or for suppressing high-frequency components while maintaining the recovered low-frequency components.

There are three advantages in using this new approach [12]: 1) the number of iterations, as well as the characteristics of

0278–0062/93$03.00 © 1993 IEEE

the filtering, can be predetermined before the iteration starts; 2) the computation time and storage space are greatly reduced as compared to TIF; and 3) once a filter is designed from two shaping matrices, a different filter can be obtained by simply changing the gain factor in the generalized Landweber iteration, or by using the dc-suppression procedure [13]. Therefore, by changing the gain factor, a spectrum of filters can be derived and investigated off-line to find out which provides the most suitable solution to the image reconstruction problem.

This paper is organized as follows. Section II formulates the image reconstruction problem in emission tomography as a problem of solving a linear system of equations, reviews TIF, and also reviews two alternative iterative methods for solving a system of linear equations: algebraic reconstruction technique (ART) and conjugate gradient (CG) method. Some drawbacks of these methods will be discussed. Section III summarizes the generalized Landweber iteration and its convergence control and recover-and-stay property. A formal proof of the recover-and-stay property, as well as the design of a shaping matrix, are also discussed and presented. This section complements original work on the generalized Landweber iteration by Strand [11] in 1974; the reader is also encouraged to read [11]. Section IV presents the new variable shaping matrix method. Two examples of using the method to accelerate the reconstruction of high-frequency components, and to attenuate the high-frequency components in the image to achieve regularization, are presented separately. The flexibility of filtering in the generalized Landweber iteration by changing the gain factor or using the dc-suppression procedure is also discussed. A comparison of computational requirements between the variable shaping matrix method and TIF is also made. Finally, Section V concludes the paper with a summary and discussion of possible applications of the variable shaping matrix to other signal processing problems.

II. Problem Formulation and Background

A. System Equation

The image reconstruction problem in emission tomography can be formulated [1], [2] as the solution of a linear system of equations represented as

$$Ax = b \tag{1}$$

where A is an $m \times n$ system matrix which describes the system geometry, x is an $n \times 1$ vector of the image pixels, and b is an $m \times 1$ vector of the measured projections of the image.

The problem is to *determine* x given A and b, a typical task of solving a linear system of equations. An obvious solution to (1) is the pseudoinverse or minimum-norm least-squares solution

$$x^* = (A^T A)^\dagger A^T b \tag{2}$$

where A^T is the matrix transpose of A and $(A^T A)^\dagger$ is the pseudoinverse of $A^T A$ (which may not have full rank). The system matrix A, in addition to being large [14], is normally ill-conditioned, so that a small perturbation in b can lead to a large change in x^*. Therefore, x^* is an unsatisfactory solution if there is noise in the measured projection data b. The ill-conditioning is due to the presence of high-frequency components of x^*, on the singular vectors with small singular values of A [9], [10].

B. Truncated Inverse Filtering

Truncated inverse filtering (TIF) [8] is used to derive a stable solution of (1) by removing high-frequency components of x^*, which are very sensitive to the measurement noise in b. A TIF solution is found as follows. First, compute the SVD of A: $A = U\Sigma V^T$. Here $U = [u_1, u_2, \cdots, u_m]$ and $V = [v_1, v_2, \cdots, v_n]$ are orthogonal matrices, and Σ is "diagonal" in that $\langle\Sigma\rangle_{i,j} = 0$ unless $i = j$, in which case $\langle\Sigma\rangle_{i,i} = \sigma_i$, the singular values of A. Without loss of generality, let $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{\rho(A)} > 0$, where $\rho(A)$ is the rank of A. Second, choose a threshold σ_T and discard the image components on singular vectors associated with singular values $\sigma_i < \sigma_T$. The number of image components kept will be $k < \min(m, n)$. Finally, form the TIF solution using

$$x^{\text{TIF}} = \sum_{i=1}^{k} (1/\sigma_i)(b, u_i)v_i \tag{3}$$

where $(b, u_i) = b^T u_i$. This aproach is straightforward if the size of A is small. However, in image reconstruction in emission tomography, the size of A will be several thousand by several thousand, and it will be even larger in 3-dimensional image reconstruction. The amount of computation and memory required to implement (3) can become enormous (see Section IV-D).

C. Alternative Iterative Approaches

Suppose the scatter can be corrected sufficiently on the measured projections b. Then matrix A is sparse in emission tomography [14]; an iterative method may be preferable for solving (1). Among iterative approaches [15] to solve (1), the algebraic reconstruction technique (ART) [1], [3], [16] and the conjugate gradient (CG) method [15], [17] have received the most attention, for their fast speed in deriving an approximate solution to (1). ART is a technique of projection onto convex set (POCS). By describing each individual linear equation in (1) as a hyperplane (also a convex set) and projecting a previous estimate from one hyperplane onto another hyperplane, a sequence of approximate solutions can be derived from the iteration. On the other hand, the CG method utilizes some orthogonality conditions [17]. The sequence of approximate solutions from the CG iteration is guaranteed to converge to x^* in no more than $\rho(A)$ iterations if a zero initial condition is used.

Since the sequence of approximate solutions from ART or CG depends on the projection b, neither ART nor CG can guarantee that a satisfactory result will be obtained after a given number of iterations. Moreover, the computation in ART is nonparallel [13], [1]. In actual application, both the ART and CG iterations must be stopped after a fixed number of iterations, since in emission tomography the sequence of approximate solutions tends to first approach a smooth solution

(image), then deviate from the smooth image, and eventually approach x^*, a highly noisy solution. Three problems exist in using ART and CG iterations: 1) choosing a stopping point may be very subjective; 2) the stopping point depends on the object to be reconstructed; and 3) there is no other control over the reconstruction process.

It has been shown in [13] and [18] that the generalized Landweber iteration with the dc-suppression procedure can be as fast as the ART or CG iteration. Moreover, the iteration is governed by the shaping matrix and the number of iterations. Once the shaping matrix and the number of iterations are determined, the outcome of the iteration can be viewed as a known *filtering* operation on the reconstructed image. The iteration is *system dependent,* not object dependent, and it allows control over the frequency content of the reconstructed image, which ART and CG cannot provide.

III. The Generalized Landweber Iteration

A. The Basic Algorithm

The generalized Landweber iteration has the form

$$x^{k+1} = x^k + \alpha D A^T (b - Ax^k) \tag{4}$$

where α is a gain factor usually set to $\alpha = 1/\sigma_1^2$, D is a *shaping matrix* (a polynomial function of $\alpha A^T A$) and x^k is the reconstructed image after the kth iteration. Here σ_1 is the largest singular value. For convenience, we define the multiplication of a vector by A as a forward projection, and the multiplication of a vector by A^T as a backward projection. In analyzing the computational efficiency of an iterative method, the number of forward and backward projections usually serves as a better indicator than the number of iterations does [13]. If the order of the polynomial function used to represent D is l, then each single generalized Landweber iteration will need $(l+1)$ forward and backward projections [18].

When $D = I$, the generalized Landweber iteration becomes the Landweber iteration [19]. When initialized with zero, the iteration converges to x^* [18], [20], provided the Euclidean norm $\|\alpha D A^T A\|_2 < 2$ [11], [21]. It has been shown that the convergence of the generalized Landweber iteration can be accelerated by using a dc-suppression procedure [13], which makes it possible to use a bigger gain $\alpha = 1/\sigma_2^2$ after a single Landweber iteration.

One example of D is $D = F(\alpha A^T A)$, where the polynomial function $F(\cdot)$ is chosen to be [11]

$$\begin{aligned} F(\lambda) = {} & 31.5 - 315\lambda + 1443.75\lambda^2 - 3465\lambda^3 \\ & + 4504.5\lambda^4 - 3003\lambda^5 + 804.375\lambda^6. \end{aligned} \tag{5}$$

This choice of polynomial function $F(\lambda)$ is made because $\lambda F(\lambda)$ is a good approximation to the unit step function in the range $\lambda \in (0, 1]$. This covers the entire spectrum of frequency components from the highest frequency component ($\sigma \neq 0$) to the lowest frequency component ($\sigma = 1$) after setting $\alpha = 1/\sigma_1^2$.

Fig. 1 shows the filtering effect of the generalized Landweber iteration if $D = F(\alpha A^T A)$ as in (5). It is clear that after about 3 iterations, the filter does not change much from one iteration to another, and that the filtering gradually changes from low-pass to all-pass as the number of iterations increases.

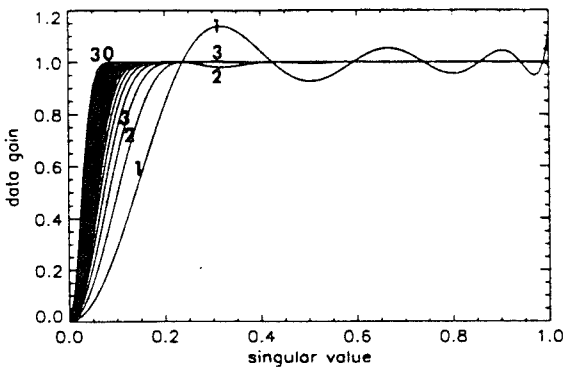


Fig. 1. The first 30 filters from the first 30 generalized Landweber iterations using the shaping matrix as specified in (5) and with $\alpha = 1/\sigma_1^2$.

In the generalized Landweber iteration, σ_1 is usually assumed to be 1 [11], although in general $\sigma_1 \neq 1$ for any given system matrix A. To effectively make $\sigma_1 = 1$, one can replace A with A divided by σ_1, the maximum singular value of A; projection data b must also be replaced by b divided by σ_1 [see (1)]. Scaling all the singular values of A by σ_1 forces all singular values of A to lie between 0 and 1.

To implement the generalized Landweber iteration, one can simply set $\alpha = 1/\sigma_1^2$ in (4) to effectively make $\sigma_1 = 1$ and ensure the convergence of iteration. It has been shown [13] that σ_1 can be obtained using the power method [22] with a few forward and backward projections. We now discuss some properties of the generalized Landweber iteration with $\sigma_1 = 1$ (which leads to $\alpha = 1$), as in [11] and [18].

B. Convergence Control

In the generalized Landweber iteration, D is an operator mapping $R(A^T)$ to $\mathcal{R}^n$, a real vector space of dimension n. Since $R(A^T)$ is spanned by $\{v_1, \cdots, v_{\rho(A)}\}$, the definition of D will be complete if the image Dv_i of each singular vector v_i in $R(A^T)$ is specified [11]. Strand [11] proposed that matrix D be designed by specifying the scalars $p_1, \cdots, p_{\rho(A)}$ in

$$Dv_i = p_i v_i, \qquad 0 < p_i \sigma_i^2 < 2 \tag{6}$$

where the condition $0 < p_i\sigma_i^2 < 2$ ensures the convergence, and will be referred to as the *convergence criterion* of the generalized Landweber iteration.

Suppose that the initial condition is x^0 and that $\sigma_1 = 1$ and $\alpha = (q/\sigma_1)^2 = q^2$. The reconstructed image x^k after the kth generalized Landweber iteration can be represented as [18]

$$\begin{aligned} x^k = {} & \sum_{i=1}^{\rho(A)} \underbrace{[1 - (1 - (p_i q^2)\sigma_i^2)^k]}_{\text{data gain}} (1/\sigma_i)(b, u_i) v_i \\ & + \sum_{i=1}^{\rho(A)} \underbrace{(1 - (p_i q^2)\sigma_i^2)^k}_{\text{initial condition gain}} (x_0, v_i) v_i. \end{aligned} \tag{7}$$

Note that the data gain and the initial condition gain, after k iterations, are governed by the factor $(1 - (p_i q^2)\sigma_i^2)^k$, which illustrates the control over convergence in the generalized Landweber iteration. By varying p_i and q, it is possible to

control the convergence rate of each component of the image independently. And, by varying k, the number of iterations, the extent of convergence can be controlled—some components of the image can be partially filtered by stopping the iteration early.

Recall that high-frequency image components are defined as the components on singular vectors associated with small singular values; low-frequency image components are associated with large singular values. It is important to note that if α in (4) is less than $1/\sigma_1^2$, say, $\alpha = 1/\sigma_>^2$, where $\sigma_>$ is greater than σ_1, i.e., $q < 1.0$ in (7), then the data gain in (7) becomes smaller than it is with $q = 1$. This implies that the data gain for each singular value becomes smaller and the convergence of the generalized Landweber iteration is slower.

On the other hand, if we use $\alpha = 1/\sigma_<^2$, where $\sigma_<$ is smaller than σ_1, i.e., $q > 1.0$ in (4), then the data gain in (7) becomes larger than it is with $q = 1.0$. It has been shown in [13] that doing this can endanger the stability of the generalized Landweber iteration. This motivated the dc-suppression procedure of [13] to set $\alpha = 1/\sigma_2^2$ after a single Landweber iteration. Since most of the singular values in this approach are equivalently shifted toward 1, the speed of the generalized Landweber iteration will be accelerated.

C. Design of a Shaping Matrix

In order to have the reconstructed image converge faster to the minimum-norm least-squares solution x^*, we need to design a polynomial function $F(\lambda)$ such that $P(\lambda) = \lambda F(\lambda)$ is as close to one as possible for $\lambda \in (0, 1]$. Note that $P(\lambda)$ is a continuous function and $P(\lambda = 0) = 0$ no matter what $F(\lambda)$ is used. Thus, the choice of $P(\lambda)$ in designing a shaping matrix will have a very limited effect on the reconstruction of the high-frequency components, which have singular values very close to 0.

Suppose $P(\lambda)$ is lth order, i.e.,

$$P(\lambda) = \lambda F(\lambda) = c_1\lambda^1 + c_2\lambda^2 + \cdots + c_l\lambda^l \qquad (8)$$

where $c_1, c_2, \cdots, c_l$ are scalars. The least squares criterion [11]

$$\text{minimize} \int_0^a P^2(\lambda)d\lambda + \int_a^1 [1 - P(\lambda)]^2 d\lambda \qquad (9)$$

is used to design $P(\lambda)$, where a is a parameter indicating a *cutoff value* between 0 and 1 for designing $P(\lambda)$. Intuitively, (9) specifies a shaping matrix $D = F(\alpha A^T A)$ such that frequency components with singular values between 0 and a will be recovered slowly, while components with singular values between a and 1 will be recovered quickly. If $a > 0$, the resulting shaping matrix will attenuate the frequency components with singular values between 0 and a. Since $P(\lambda)$ is continuous and $P(0) = 0$, this approach shows little success [11].

Minimizing (9) with given order l and cutoff value a leads to the following set of linear equations:

$$\sum_{i=1}^{l} \frac{c_i}{i+j+1} = \frac{1-a^{j+1}}{j+1}, \qquad j = 1, \cdots, l. \qquad (10)$$

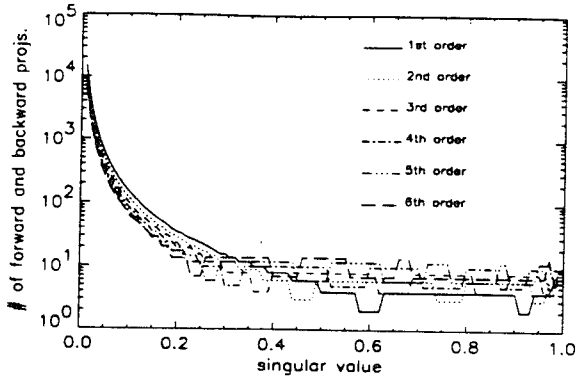


Fig. 2. The number of iterations required to recover 95% of the object components projected onto singular vectors associated with various singular values in the generalized Landweber iteration using $F_l(\lambda), l = 1, \cdots, 6$. Discontinuous structure arises because there are $l + 1$ forward and backward projections per complete generalized Landweber iteration.

Several examples of polynomial functions $F_l(\lambda)$ from (10) with $a = 0$ and orders $l = 1, \cdots, 6$ are listed as follows:

$$F_1(\lambda) = 4 - 3\frac{1}{3}\lambda$$
$$F_2(\lambda) = 7.5 - 15\lambda + 8.75\lambda^2$$
$$F_3(\lambda) = 12 - 42\lambda + 56\lambda^2 - 25.2\lambda^3$$
$$F_4(\lambda) = 17.5 - 93\frac{1}{3}\lambda + 210\lambda^2 - 210\lambda^3 + 77\lambda^4$$
$$F_5(\lambda) = 24 - 180\lambda + 600\lambda^2 - 990\lambda^3 + 792\lambda^4 - 245.14286\lambda^5$$
$$F_6(\lambda) = 31.5 - 315\lambda + 1443.75\lambda^2 - 3465\lambda^3 + 4504.5\lambda^4 - 3003\lambda^5 + 804.375\lambda^6.$$

Note that $F_6(\lambda)$ is the same as $F(\lambda)$ in (5).

Fig. 2 shows the number of iterations required to recover 95 percent of the object components projected onto singular vectors associated with various singular values in the generalized Landweber iteration for each of the above 6 polynomial functions. It is clear that using a higher order polynomial function results in a faster reconstruction of high-frequency components ($\sigma < 0.2$), at the expense of slower reconstruction of low-frequency components ($\sigma > 0.2$). After about 20 forward and backward projections, a higher order polynomial function outperforms a lower order polynomial function in the sense that the low-frequency components with $\sigma > 0.2$ have been almost completely recovered, and the high-order polynomial function has recovered more of the high-frequency components with $\sigma < 0.2$.

Moving the value a away from 0 may also result in designing a shaping filter with negative gains in the high-frequency range associated with small singular values. Combining the positive and negative gains in a range of frequency components (usually in high-frequency range) can result in cancellation of the frequency components in this range. Since signal-to-noise ratio is normally small in the high-frequency range, the high-frequency components (both the reconstructed image and the reconstructed noise) may cause the image to look noisy. Removing the high-frequency components can result in a smoother reconstructed image.

D. Recover-and-Stay Property

We now state the *recover-and-stay* property: image components, once recovered within some ϵ, will still stay within ϵ in the subsequent iterations even if D is changed to D', as long as D' satisfies the convergence criterion for those almost recovered components. A formal proof of this property is as follows.

Given zero initial condition ($x^0 = 0$), from (7) the reconstructed image x^{k_1} after k_1 generalized Landweber iterations is

$$x^{k_1} = \sum_{i=1}^{\rho(A)} [1 - (1 - p_i\sigma_i^2)^{k_1}](1/\sigma_i)(b, u_i)v_i. \qquad (11)$$

Suppose the first l components have been reconstructed to within a factor ϵ in x^{k_1}, i.e.,

$$|(1 - p_i\sigma_i^2)^{k_1}| < \epsilon, \qquad i = 1, \cdots, l. \qquad (12)$$

Then even with a *different shaping matrix* D' such that $D'v_i = p_i'v_i$, the reconstructed image is (from (7) with $x_0 = x^{k_1}$)

$$\begin{aligned} x^k &= \sum_{i=1}^{\rho(A)} [1 - (1 - p_i'\sigma_i^2)^k](1/\sigma_i)(b, u_i)v_i \\ &\quad + \sum_{i=1}^{\rho(A)} (1 - p_i'\sigma_i^2)^k (x^{k_1}, v_i)v_i \\ &= \sum_{i=1}^{\rho(A)} [1 - (1 - p_i'\sigma_i^2)^k](1/\sigma_i)(b, u_i)v_i \\ &\quad + \sum_{i=1}^{l} (1 - p_i'\sigma_i^2)^k (x^{k_1}, v_i)v_i \\ &\quad + \sum_{i=l+1}^{\rho(A)} (1 - p_i'\sigma_i^2)^k (x^{k_1}, v_i)v_i \\ &= \sum_{i=1}^{l} [1 - (1 - p_i'\sigma_i^2)^k (1 - p_i\sigma_i^2)^{k_1}](1/\sigma_i)(b, u_i)v_i \\ &\quad + \sum_{i=l+1}^{\rho(A)} [1 - (1 - p_i'\sigma_i^2)^k](1/\sigma_i)(b, u_i)v_i \\ &\quad + \sum_{i=l+1}^{\rho(A)} (1 - p_i'\sigma_i^2)^k (x^{k_1}, v_i)v_i. \end{aligned} \qquad (13)$$

If D' satisfies the convergence criterion in (6) for the first l components, then

$$|(1 - p_i'\sigma_i^2)^k| < 1, \qquad i = 1, \cdots, l. \qquad (14)$$

The coefficients in the first summation of the last equation of (13) will then satisfy

$$1 - \epsilon < 1 - (1 - p_i'\sigma_i^2)^k (1 - p_i\sigma_i^2)^{k_1} < 1 + \epsilon, \qquad i = 1, \cdots, l. \qquad (15)$$

As long as D', the second shaping matrix, satisfies the convergence criterion in (6) for the almost recovered components, the maximum deviation for these components will still be within ϵ in the subsequent iterations with D'. We will show by examples that it is not difficult to have $\epsilon < 0.05$. This proves the recover-and-stay property of the generalized Landweber iteration.

IV. The Variable Shaping Matrix Method

From the recover-and-stay property, different shaping matrices may be used to emphasize or deemphasize some frequency components. We have found that one shaping matrix can be used for the first several iterations to recover low-frequency components of an image; and then another shaping matrix, which requires less computation than the first shaping matrix in a single generalized Landweber iteration, can be used to accelerate the reconstruction of high-frequency components. We also found that if the second shaping matrix provides negative gains in a range of high-frequency components, the partial recovery of the high-frequency components in this range using the first shaping matrix can be removed. In order to remove some components, from the proof of the recover-and-stay property, the second shaping matrix has to violate the convergence criterion for the components to be removed. The result of removing high-frequency components is very similar to the result using TIF. Two different examples are provided to demonstrate 1) acceleration of convergence, and 2) attenuation of high-frequency components.

A. Acceleration of Reconstruction of High-Frequency Components

Suppose the shaping matrix $D = F(\alpha A^T A)$, where $F(\cdot)$ is defined in (5), is used in the first three iterations of the generalized Landweber iteration with $\alpha = 1/\sigma_1^2$. From Fig. 1, we know that the frequency components with singular values above 0.2 are recovered up to 95 percent ($\epsilon < 0.05$) in these three iterations (which are equivalent to 21 forward and backward projections).

Starting with the fourth iteration, suppose the iteration switches the shaping matrix $D = F(\alpha A^T A)$ to $D' = F'(\alpha A^T A)$, where $F'(\cdot)$ is defined as

$$\begin{aligned} F'(\lambda) &= 25.94897 - 126.47433\lambda \\ &\quad + 200.73822\lambda^2 - 100.03043\lambda^3. \end{aligned} \qquad (16)$$

$F'(\lambda)$ is obtained from the Chebyshev approximation [23] of a third-order polynomial to $F(\lambda)$ in (5). The idea is to use a lower order polynomial to approximate $F(\lambda)$ to maintain the response in the range of high-frequency components, while relaxing the response in the range of low-frequency components, which have almost been recovered.

Fig. 3 shows the convergence properties of (16) for the first 30 generalized Landweber iterations. Although there are large variations from one iteration to another iteration in the range of singular values [0.2,1.0], the performance in the range from 0 to 0.2 is very similar to that of using (5) in Fig. 1. Moreover, D' still satisfies the convergence criterion for $\sigma \in [0.2,1.0]$ (see Fig. 3).

Fig. 4 shows that $F'(\lambda)$ actually outperforms $F(\lambda)$ in recovering 95 percent of the high-frequency components with singular values $\sigma \in (0, 0.2]$. Although $F'(\lambda)$ behaves poorly when $\sigma \in [0.2, 1.0]$, from the recover-and-stay property, $F'(\lambda)$

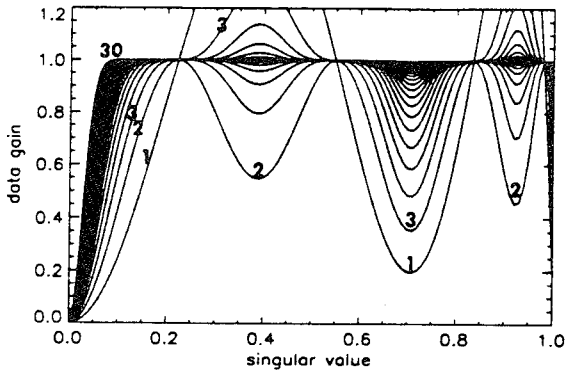


Fig. 3. Convergence properties of the third-order polynomial function $F'(\lambda)$ for the first 30 generalized Landweber iterations. The first, second, third, and final iterations are labeled.

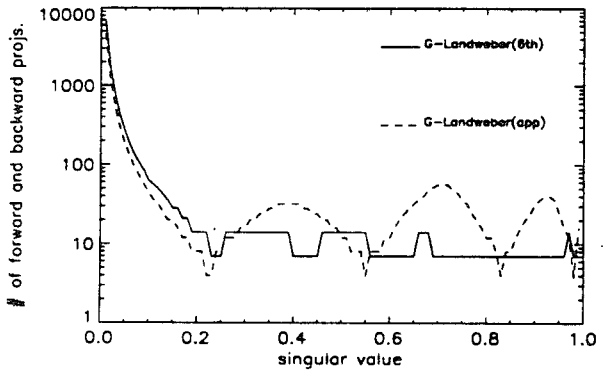


Fig. 4. Comparison of performance in recovering 95% of components on various singular values using the generalized Landweber iteration with $F(\lambda)$ [marked as G-Landweber(6th)], and with $F'(\lambda)$ [marked as G-Landweber(app)]. It is clear that $F'(\lambda)$ outperforms $F(\lambda)$ in $\sigma \in (0,0.2]$ since it requires less computation than $F(\lambda)$.

can be applied *after* the low-frequency components with $\sigma \in [0.2,1.0]$ have been almost recovered from using $F(\lambda)$.

We used this method to simulate the noise-free reconstruction of the complex phantom in a simulated positron emission tomography (PET) geometry. We used the PET-3 geometry in [13] and [18], whose system matrix is 4160×3228, with 4160 projection elements and 3228 image pixels. Fig. 5 compares the performance of using $F(\lambda)$ alone, and the performance of using $F(\lambda)$ for three iterations followed by using $F'(\lambda)$. This shows how the variable shaping matrix method speeds up the reconstruction of high-frequency components.

B. Attenuation of High-Frequency Components in the Reconstruction

We now demonstrate another way of using a variable shaping matrix: to attenuate high-frequency components in the reconstructed image to obtain a smooth image. This can also be viewed as a solution to a regularized image reconstruction problem. Using a shaping matrix $D = F(\alpha A^T A)$ as in (5) will partially recover some high-frequency components, which may deteriorate the image quality significantly if these components are noisy. Thus, in some applications, it may be desirable to remove high-frequency components, including both the reconstructed image and the reconstructed noise, in the range (say) $\sigma \in (0,0.1]$, to make the image smooth. We now show

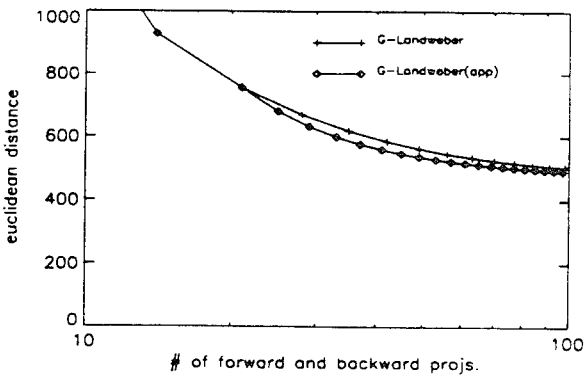


Fig. 5. Comparison between using $F(\lambda)$ alone (marked as G-Landweber) and using $F(\lambda)$ for the first three iterations followed by using $F'(\lambda)$ [marked as G-Landweber(app)]. The plus and diamond symbols represent the result of a single iteration. Note that $F'(\lambda)$ requires less computation than $F(\lambda)$ does to achieve the same Euclidean distance between the original and reconstructed images.

how the generalized Landweber iteration can do this *during* the image reconstruction process.

Suppose the polynomial function $F(\lambda)$ in (5) is used for the first three generalized Landweber iterations to ensure that low-frequency components with $\sigma \in (0.2,1]$ have been almost recovered. Fig. 1 shows the response using $F(\lambda)$ in the first three generalized Landweber iterations. Unfortunately, it is clear that high-frequency components with $\sigma \in [0,0.2]$ have also been partially recovered. We would like to avoid this.

Now suppose the polynomial function

$$F''(\lambda) = -10.972414 + 209.51804\lambda - 1072.02\lambda^2 - 2599.2093\lambda^3 - 3313.8866\lambda^4 + 2146.6546\lambda^5 - 557.55271\lambda^6 \quad (17)$$

is used for the subsequent iterations. $F''(\lambda)$ is found by designing a polynomial function [see (10)] with cutoff value $a = 0.15$ and order $l = 6$. It provides *negative* gains in the range $\sigma \in (0,0.28]$, which covers $\sigma \in (0,0.2]$. $F''(\lambda)$ is used for another nine generalized Landweber iterations to ensure that sufficient attenuation or removal of high-frequency components has been achieved. In this case, the shaping matrix $D'' = F(\alpha A^T A)$ does not satisfy the convergence criterion for the components in $\sigma \in (0,0.28]$. Fig. 6(a) shows the response using $F''(\lambda)$ alone for nine generalized Landweber iterations. Note that this iteration cannot be allowed to continue for many iterations, since the iteration will diverge.

Now, if $F(\lambda)$ is used for the first three generalized Landweber iterations, and $F''(\lambda)$ for the subsequent nine iterations, the resulting response after these twelve iterations is similar to the result of a *low-pass filtering* with a *transition band* having low cutoff frequency at $\sigma_{lo} = 0.19$ and high cutoff frequency at $\sigma_{hi} = 0.08$. Fig. 6(b) shows the transition (iteration by iteration), combining the use of $F(\lambda)$ for three iterations (solid lines) and the use of $F''(\lambda)$ for the subsequent nine iterations (dotted lines). The resulting low-pass filter after the twelve iterations has the low cutoff and high cutoff frequencies at $\sigma_{hi} = 0.08$ and $\sigma_{lo} = 0.19$, respectively. The cutoff frequencies σ_{hi} and σ_{lo} were chosen at the frequencies with gains 0.05 and 0.95, separately. In general, this type of shaping is also referred to as rolloff shaping [8]. For the components

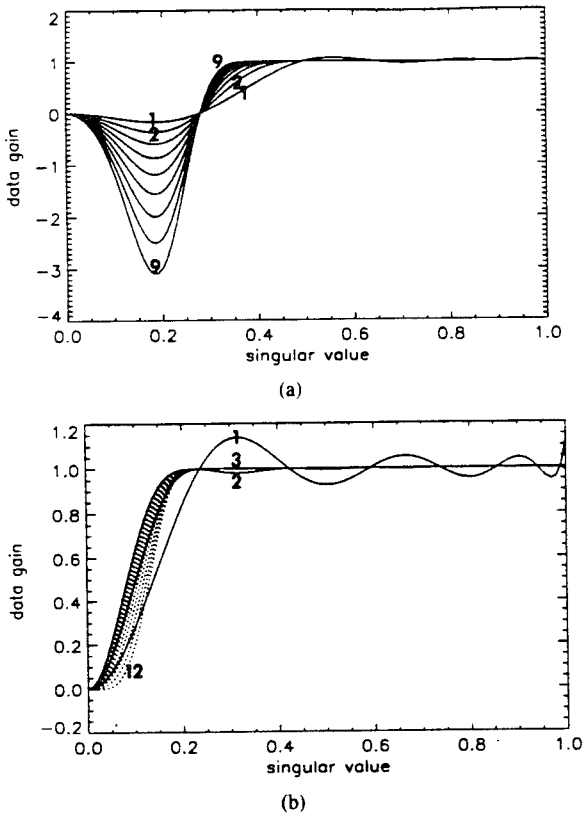


Fig. 6. (a) The response using $F''(\lambda)$ for the first nine generalized Landweber iterations. (b) The transition from combining $F(\lambda)$ for three iterations (solid lines) and $F''(\lambda)$ for the subsequent nine iterations (dotted lines). The resulting lowpass filter (rightmost dotted line) after the twelve iterations has the low cutoff and high cutoff frequencies at $\sigma_{\text{hi}} = 0.08$ and $\sigma_{\text{lo}} = 0.19$, respectively. The cutoff frequencies σ_{hi} and σ_{lo} were chosen at the frequencies with gains 0.05 and 0.95, separately.

in $\sigma \in [0.2,0.28]$, where the almost recovered components and to-be-removed components overlap, the second shaping does not affect much for the small ϵ (about less than 0.01).

We used this method to simulate the reconstruction of the complex phantom in the simulated PET-3 geometry from noisy data. The number of total counts was 1 million; the data noise was Poisson in nature. Fig. 7 shows the result of the twelve iterations. Compare the third and the twelfth reconstructed images of Fig. 7. As expected, the twelfth reconstructed image is smoother than the third image.

An important feature of the generalized Landweber iteration can also be seen in Fig. 7—not only have we obtained the reconstructed images, but we also have a *known* relation between a reconstructed image and the shaping filter generating the image. This known relation is unique to the generalized Landweber iteration; it is not present in the ART or CG iterations. Since $F(\lambda)$ and $F''(\lambda)$ are both polynomial functions of order 6, $84(=(6+1)\times 12)$ forward and backward projections are *known to be needed* to obtain the desired image.

In examining the sequence of images in Fig. 7, note the change due to the amplification and attenuation of the components in $\sigma \in [0.2,1.0]$. The deterioration of the third image was due to the partial reconstruction of high-frequency components. The change in image quality from the first image to the second image, and from the second image to the third image, shows that partial reconstruction of the high-frequency

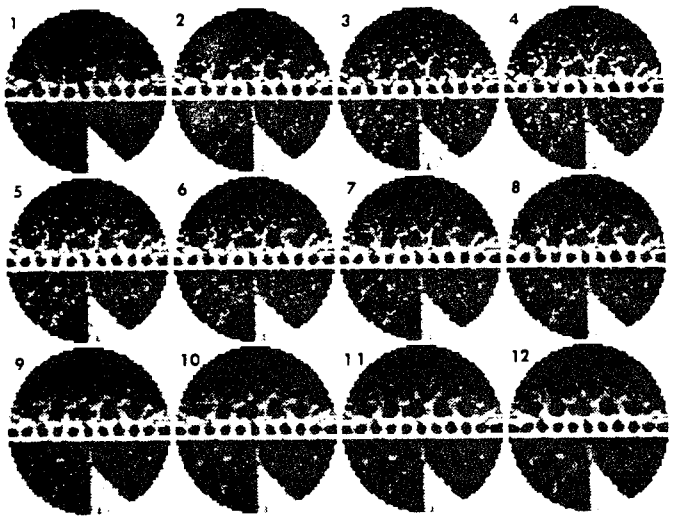


Fig. 7. The reconstructed images from the first three generalized Landweber iterations using $F(\lambda)$ and the nine subsequent iterations using $F''(\lambda)$.

components can deteriorate image quality. This shows the necessity of filtering out the high-frequency components to smooth the image.

C. A Family of Filters Based on the Same Shaping Matrix with a Different Gain Factor

We show that it is possible to use the same shaping filter with a different gain factor α in (4) to target different ranges of high-frequency components. Thus, noise filtering can be performed without designing another shaping filter.

For example, set $\alpha = 1/\sigma_1^2$ and $D = I$ during the first iteration; then adopt the dc-suppression procedure and set $\alpha = 1/\sigma_2^2$ and $D = F(\alpha A^T A)$ for the subsequent iterations. The low cutoff and high cutoff transition band frequencies will change from σ_{lo} and σ_{hi} to $(\sigma_2/\sigma_1)\sigma_{\text{lo}}$ and $(\sigma_2/\sigma_1)\sigma_{\text{hi}}$, respectively; the low cutoff σ_{lo} and high cutoff σ_{hi} frequencies are both multiplied by the ratio σ_2/σ_1. In our example, $\sigma_1 = 0.986703$ and $\sigma_2 = 0.587256$. If $\alpha = (q/\sigma_1)^2$, where $q \leq 1$, then the low cutoff and high cutoff transition band frequencies become $(1/q)\sigma_{\text{lo}}$ and $(1/q)\sigma_{\text{hi}}$, respectively.

Fig. 8 shows the frequency responses after three generalized Landweber iterations, using the polynomial function $F(\lambda)$ in (5), with the gain factor α set equal to: $1/\sigma_2^2$, $1/\sigma_1^2$, $(0.8/\sigma_1)^2$, $(0.7/\sigma_1)^2$, $(0.6/\sigma_1)^2$, $(0.5/\sigma_1)^2$, and $(0.4/\sigma_1)^2$, separately. Fig. 9 shows the frequency responses and cutoff frequencies of using the resulting filter after 12 iterations in Fig. 6(b), with the gain factor α set equal to $1/\sigma_2^2$, $1/\sigma_1^2$, $(0.8/\sigma_1)^2$, $(0.7/\sigma_1)^2$, $(0.6/\sigma_1)^2$, $(0.5/\sigma_1)^2$, and $(0.4/\sigma_1)^2$, separately. We call the shaping filters in Fig. 8 *unregularized* filters, and the shaping filters in Fig. 9 *regularized* filters.

The reconstructed images using the unregularized filters in Fig. 8 and the regularized filters in Fig. 9 are shown in Figs. 10 and 11, respectively. The number at the upper-left corner of each image corresponds to the filter with the same number. The number 0 indicates the original image. As expected, the images in Fig. 11 are smoother than the corresponding images in Fig. 10. One important extension of using a family of shaping filters is to allow us to choose an appropriate filter in the image reconstruction.

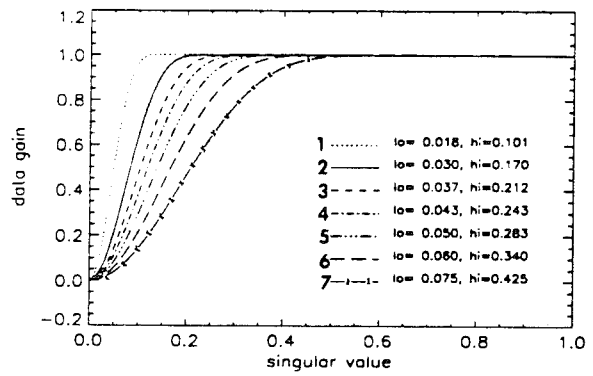


Fig. 8. The frequency responses after the first three generalized Landweber iterations using $F(\lambda)$ with $\alpha = 1/\sigma_2^2$, $1/\sigma_1^2$, $(0.8/\sigma_1)^2$, $(0.7/\sigma_1)^2$, $(0.6/\sigma_1)^2$, $(0.5/\sigma_1)^2$, and $(0.4/\sigma_1)^2$, separately.

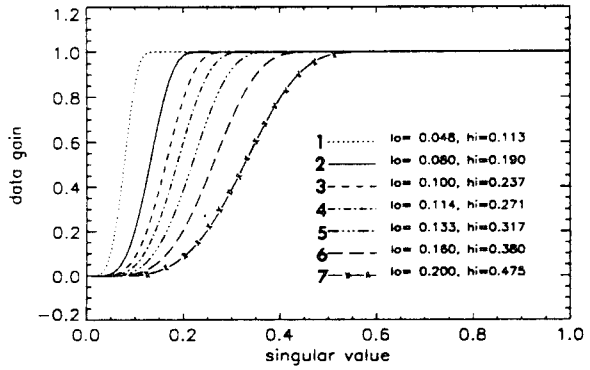


Fig. 9. The frequency responses using the resulting filter with $\alpha = 1/\sigma_2^2$, $1/\sigma_1^2$, $(0.8/\sigma_1)^2$, $(0.7/\sigma_1)^2$, $(0.6/\sigma_1)^2$, $(0.5/\sigma_1)^2$, and $(0.4/\sigma_1)^2$, separately.

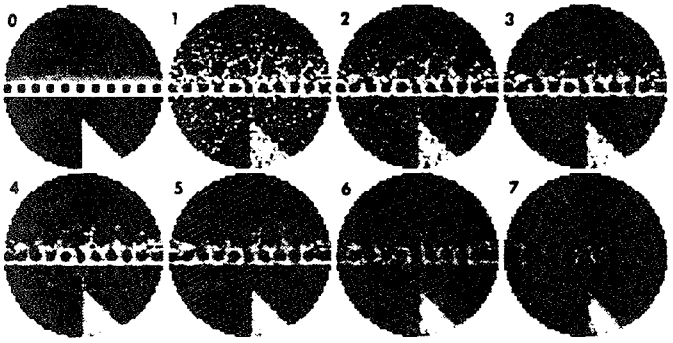


Fig. 10. The reconstructed images from the unregularized filters.

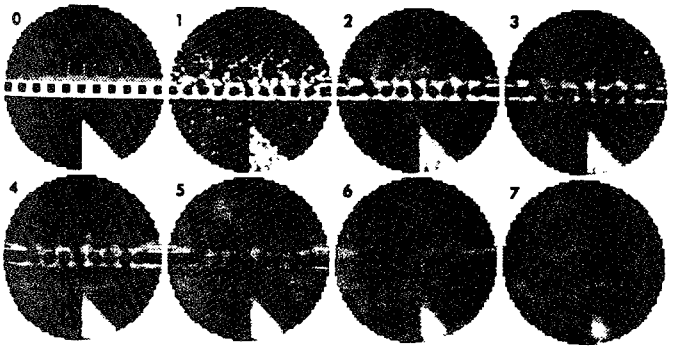


Fig. 11. The reconstructed images from the regularized filters.

D. Comparison of Computational Requirement of Rolloff Shaping and Truncated Shaping

We compare the computational requirements for a rolloff shaping filter (see Section IV-B) and a truncated shaping filter as in TIF. The major difference is that the rolloff shaping can be iteratively performed (84 forward and backward projections for the shapings in Fig. 9), while the truncated shaping must be computed directly using (2); this requires an enormous amount of computation.

Direct computation of x^* using (3) requires storage of the matrices U and V and a considerable amount of computation to implement (3), since U and V (unlike A) are not sparse. It also requires the (off-line) computation of the SVD of A. Consider a 4000 × 4000 system matrix A, and assume using 4 bytes to store a number in floating-point format. Then 128 million (= 4000 × 4000 × 4 × 2) bytes will be required just to store U and V! On the other hand, if A is 97 percent sparse (3 percent of the elements of A are nonzero), the computation time for obtaining x^* in (3) is equivalent to about 33($\approx$2/0.06) forward and backward projections when the sparse structure of A is utilized in the iteration. If k in (3) is chosen to preserve half of the singular values, then 17 forward and backward projections are required for the computation.

In some applications, the system matrix A without modeling scatter becomes sparser as its size increases. For example, a 150-million-elements system matrix in [14] has about 2 million nonzero elements, so that it is 98.7 percent sparse. In this case, an SVD computation will become almost impossible. The computation time for obtaining x^* is equivalent to about 75($\approx$2/0.026) forward and backward projections. If half of the singular values are preserved, about 37 forward and backward projections will be needed. Hence, *even apart from the storage problem and the difficulty in computing the SVD of A*, the iterative approach of rolloff shaping becomes more favorable over the direct approach using TIF as the size and sparseness of A grows.

V. Summary

A new variable shaping matrix method for using the generalized Landweber iteration in image reconstruction in emission tomography has been developed. The method uses two shaping matrices: the first one is for fast recovery of low-frequency components; and the second one is either for the acceleration of the reconstruction of high-frequency components if the noise does not impose a problem in recovering the high-frequency components, or for the attenuation of high-frequency components when noise is a problem. The recover-and-stay property of the generalized Landweber iteration has been stated and proved for the first time in this paper; it serves as the foundation of the variable shaping matrix method.

Examples of using a variable shaping matrix to accelerate reconstruction and to attenuate high-frequency components were presented. A sequence of filters from the same shaping matrices with different gain factors were also shown. From this sequence, one may choose an appropriate filter for the image reconstruction. The savings in computation and storage space in using a variable shaping matrix instead of TIF was also

presented; it results from the sparseness of A, which generally increases as the size of A increases. Hence, the larger the system, the greater the advantage of the new method over TIF.

An important topic for further research is application of the generalized Landweber iteration with a variable shaping matrix to linear-algebra-based signal processing problems. The solution (2) is used in signal restoration problems, harmonic retrieval, and deconvolution. The method proposed in this paper can be applied to all of those problems, even if the matrix is not sparse; the computation required is less than that for an SVD. Another topic is further study on designing the shaping matrix to achieve a given specification of low and high cutoff frequencies with the least amount of computation.

ACKNOWLEDGMENT

The authors would like to thank the reviewers for their very constructive suggestions.

REFERENCES

[1] Y. Censor, "Finite series-expansion reconstruction methods," *IEEE Proc.*, vol. 71, pp. 409–419, Mar. 1983.

[2] L. A. Shepp and Y. Vardi, "Maximum likelihood reconstruction for emission tomography," *IEEE Trans. Med. Imaging*, vol. MI-1, pp. 113–122, Oct. 1982.

[3] G. T. Herman, A. Lent, and S. W. Rowland, "ART: Mathematics and applications," *J. Theor. Biol.*, vol. 42, pp. 1–32, 1973.

[4] J. Llacer, "Tomographic image reconstruction by eigenvector decomposition: its limitations and areas of applicability," *IEEE Trans. Med. Imaging*, vol. MI-1, pp. 34–42, July 1982.

[5] U. Raff, D. N. Stroud, and W. R. Hendee, "Improvement of lesion detection in scintigraphic images by SVD techniques for resolution recovery," *IEEE Trans. Med. Imaging*, vol. MI-5, pp. 35–44, Mar. 1986.

[6] H. H. Barrett, J. N. Aarsvold, and T. J. Roney, "Null functions and eigenfunctions: Tools for analysis of imaging systems," in *Information Processing in Medical Imaging*, pp. 211–226, 1991.

[7] M. F. Smith, C. E. Floyd, and R. E. Coleman, "Reconstruction of SPECT images using generalized matrix inverses," in *Proc. Symp. IEEE Nucl. Sci.*, Arlington, VA, 1990.

[8] C. K. Rushforth, "Signal restoration, functional analysis, and Fredholm integral equations of the first kind," in *Image Recovery: Theory and Applications*, H. Stark, ed. Orlando, FL: Academic, 1987, ch. 1, pp. 1–27.

[9] G. Wahba, "Three topics in ill-posed problems," in *Inverse and Ill-Posed Problems*, H. W. Engl and C. W. Groetsch, eds. Orlando, FL: Academic, 1987, pp. 37–51.

[10] G. Demoment, "Image reconstruction and restoration: overview of common estimation structures and problems," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-37, pp. 2024–2036, Dec. 1989.

[11] O. N. Strand, "Theory and methods related to the singular-function expansion and Landweber's iteration for integral equations of the first kind," *SIAM J. Numer. Anal.*, vol. 11, no. 4, pp. 798–825, Sept. 1974.

[12] T.-S. Pan, "The generalized Landweber iteration in positron emission tomography," Ph.D. dissertation, Univ. of Michigan, 1991.

[13] T.-S. Pan and A. E. Yagle, "Acceleration of Landweber-type algorithms by suppression of projection on the maximum singular vector," *IEEE Trans. Med. Imaging*, vol. 16, pp. 479–487, Dec. 1992.

[14] L. Kaufman, "Implementing and accelerating the EM algorithm for positron emission tomography," *IEEE Trans. Med. Imaging*, vol. MI-6, pp. 37–51, Mar. 1987.

[15] T. F. Budinger, G. T. Gullberg, and R. H. Huesman, "Emission computed tomography," in *Reconstruction from Projections: Implementation and Applications*, G. T. Herman, ed. New York: Springer-Verlag, 1979, ch. 5, pp. 147–246.

[16] G. T. Herman and A. Lent, "Iterative reconstruction algorithms," *Comput. Biol. Med.*, vol. 6, pp. 273–294, 1976.

[17] S. Kawata and O. Nalcioglu, "Constrained iterative reconstruction by the conjugate gradient method," *IEEE Trans. Med. Imaging*, vol. MI-4, pp. 65–71, June 1985.

[18] T.-S. Pan and A. E. Yagle, "Numerical study of multigrid implementations of some iterative image reconstruction algorithms," *IEEE Trans. Med. Imaging*, vol. MI-10, pp. 572–588, Dec. 1991.

[19] L. Landweber, "An iterative formula for Fredholm integral equations of the first kind," *Amer. J. Math.*, vol. 73, pp. 615–624, 1951.

[20] D. G. Luenberger, *Optimization by Vector Space Methods*. New York: Wiley, 1968.

[21] J. L. C. Sanz and T. S. Huang, "Unified Hilbert space approach to iterative least-squares linear signal restoration," *J. Opt. Soc. Amer.*, vol. 73, no. 11, pp. 1455–1465, Nov. 1983.

[22] A. Jennings, *Matrix Computation for Engineers and Scientists*. New York: Wiley, 1977.

[23] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes in C*. New York: Cambridge University Press, 1988.

APPENDIX F2

T.-S. Pan and A.E. Yagle, "Acceleration and Filtering in the Generalized Landweber Iteration using a Variable Shaping Matrix," IEEE 1991 Medical Imaging Conference, Santa Fe, NM, Nov. 5-9, 1991, pp. 2028-2032.

This is the conference paper version of Appendix F1.

Acceleration and Filtering in the Generalized Landweber Iteration Using a Variable Shaping Matrix

Tin-Su Pan[†] *, Andrew E. Yagle[†], Neal H. Clinthorne[‡] and W. Leslie Rogers[‡]
[†]Department of Electrical Engineering and Computer Science
The University of Michigan, Ann Arbor, MI 48109-2122
[‡]Division of Nuclear Medicine, University of Michigan Medical Center

ABSTRACT

We use a variable shaping matrix in the generalized Landweber iteration to solve the large linear system of equations arising in the image reconstruction problem of emission tomography. Our method is based on the property that once a spatial frequency image component is recovered in the generalized Landweber iteration, this component will not change during subsequent iterations, even if a different shaping matrix is used. Two different shaping matrices are used: the first recovers low-frequency image components; and the second may be used either to accelerate the reconstruction of high-frequency image components, or to attenuate these components to filter the image. The variable shaping matrix gives results similar to truncated inverse filtering, but requires much less computation and memory, since it does not rely on the singular value decomposition.

I. INTRODUCTION

In emission tomography, the image reconstruction problem can be formulated as the solution to a large linear system of equations. This system of equations tends to be large, with dimensions on the order of thousands, and ill-conditioned. One approach to this problem is to perform a singular value decomposition (SVD) of the system matrix and set some small singular values to zero to obtain a stable (not sensitive to the measurement noise) solution. This approach is called truncated inverse filtering (TIF) [1].

This paper defines "spatial frequencies" as the reciprocals of singular values, and the associated image components as the projection of the image on the corresponding singular vectors. High spatial frequency components of the object are defined as components on singular vectors associated with small singular values; and low spatial frequency components as components on singular vectors with large singular values. Using this terminology in TIF, high frequency components of the reconstructed image, which are sensitive to noise, will be removed.

To compute the SVD of a large system matrix will take a long time and much memory space. Furthermore, in emission tomography, attenuation results in different objects having different system matrices, each of which would require a SVD. Thus using TIF to derive a solution for the image reconstruction in emission tomography requires a huge amount of computation and memory space.

This paper presents a novel and alternative approach – using the generalized Landweber iteration [2] with a variable shaping matrix (defined below). This method can either 1) accelerate the reconstruction of high frequency components; or 2) roll off the inverse filter to prevent Gibbs phenomenon arising from the sharp truncation in TIF, and preserve the stability of the solution when the system matrix is ill-conditioned. The motivation for using a variable shaping matrix is based on the recover-and-stay property of the generalized Landweber iteration which is stated and proved in [3]. This property states that *in the generalized Landweber iteration, once an image component is completely recovered, the component will not change during further iterations, even if a different shaping matrix is used afterwards.* Our new approach uses two different shaping matrices: the first one is for fast recovery of low frequency components and partial recovery of high frequency components; the second one may be used either for speeding up the reconstruction of high frequency components (if noise is not a problem), or for suppressing high frequency components while maintaining the recovered low frequency components.

There are three advantages in using this new approach [3]: 1) the number of iterations, as well as the characteristics of the filtering, can be predetermined before the iteration starts; 2) the computation time and storage space are greatly reduced as compared to TIF; and 3) once a filter is designed from two shaping matrices, a different filter can be obtained by simply changing the gain factor in the generalized Landweber iteration, or by using the DC-suppression procedure [3]. Therefore, by changing the gain factor, a spectrum of filters can be derived and inves-

*The work of Tin-Su Pan was supported in part by NIH grant #PO1-CA42768, and in part by a Research Partnership award from the H. H. Rackham School of Graduate Studies of the University of Michigan. The work of Andrew E. Yagle, Neal H. Clinthorne, and W. Leslie Rogers was supported in part by ONR grant #N00014-90-J-1897, NIH grant #RO1-CA32846, and NIH grant #RO1-CA54362, respectively.

0-7803-0513-2/92$0.3.00 ©IEEE

tigated off-line to find out which provides the most suitable solution to the image reconstruction problem.

This paper is organized as follows. Section II formulates the image reconstruction problem in emission tomography as a problem of solving a linear system of equations, reviews TIF and two alternative iterative methods for solving a system of linear equations: algebraic reconstruction technique (ART) and conjugate gradient (CG) method, and also reviews the generalized Landweber iteration. Section III presents the new variable shaping matrix method. Two examples of using the method to accelerate the reconstruction of high frequency components, and to attenuate the high frequency components in the image to achieve regularization, are presented separately. A comparison of computational requirements between the variable shaping matrix method and TIF is made. Finally, Section IV concludes the paper with a summary and discussion of possible applications of the variable shaping matrix to other signal processing problems.

II. Background

A. System Equation

The image reconstruction problem in emission tomography can be formulated [4] as to solve the linear system of equations represented as

$$Ax = b, \tag{1}$$

where A is an $m \times n$ system matrix which describes the system geometry, x is an $n \times 1$ vector of the image pixels, and b is an $m \times 1$ vector of the measured projections of the image.

The problem is to *determine* x given A and b, a typical task of solving a linear system of equations. An obvious solution to (1) is the pseudo-inverse or minimum-norm least-squares solution:

$$x^* = (A^T A)^\dagger A^T b, \tag{2}$$

where A^T is the matrix transpose of A and $(A^T A)^\dagger$ is the pseudo-inverse of $A^T A$ (which may not have full rank). The system matrix A, in addition to being large [5], is normally ill-conditioned, so that a large change in x^* can result from a small perturbation in b. Therefore, x^* is an unsatisfactory solution if there is noise in the measured projection data b. The ill-conditioning is due to the presence of high frequency components of x^*, on the singular vectors with small singular values of A.

B. Truncated Inverse Filtering

Truncated inverse filtering (TIF) [1] is used to derive a stable solution of (1) by removing the high frequency components of x^*, which are very sensitive to the measurement noise in b. The TIF solution is found as follows. First, compute the SVD of A: $A = U\Sigma V^T$. Here $U = [u_1, u_2, \ldots, u_m]$ and $V = [v_1, v_2, \ldots, v_n]$ are orthogonal matrices, and Σ is "diagonal" in that $<\Sigma>_{i,j} = 0$ unless $i = j$, in which case $<\Sigma>_{i,i} = \sigma_i$, the singular-values of A. Without loss of generality, let $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{\rho(A)} > 0$, where $\rho(A)$ is the rank of A. Second, choose a threshold σ_T and discard the image components on singular vectors associated with singular values $\sigma_i < \sigma_T$. The number of image components kept will be $k < m, n$. Finally, form the TIF solution using

$$x^{\mathrm{TIF}} = \sum_{i=1}^{k} \frac{1}{\sigma_i}(b, u_i)v_i, \tag{3}$$

where $(b, u_i) = b^T u_i$. This approach is straightforward if the size of A is small. However, in image reconstruction in emission tomography, the size of A will be several thousand by several thousand, and it will be even larger in 3-dimensional image reconstruction. The amount of computation and memory required to implement (3) can become enormous, even though the SVD is precomputed off-line (see subsection III.C).

C. Alternative Iterative Approaches

Since the matrix A is sparse in emission tomography, an iterative method may be preferable when solving (1). Among iterative approaches [6] to solve (1), the algebraic reconstruction technique (ART) and the conjugate gradient (CG) method have received most attention, for their fast speed in deriving an approximate solution to (1). ART is a technique of projection onto convex set (POCS). CG method utilizes some orthogonality conditions. Since the sequence of approximate solutions from ART or CG depends on the projection b, neither ART nor CG can guarantee that a satisfactory result will be obtained after a given number of iterations.

D. Generalized Landweber Iteration

It has been shown [7] that the generalized Landweber iteration with the DC-suppression procedure can be as fast as the ART or CG iterations. Moreover, the iteration is governed by the shaping matrix and the number of iterations. Once the shaping matrix and the number of iterations are determined, the outcome of the iteration can be viewed as a known *filtering* operation on the reconstructed image. The iteration is *system dependent*, not object dependent, and it allows control over the frequency content of the reconstructed image, which ART and CG cannot provide.

The generalized Landweber iteration has the form

$$x^{k+1} = x^k + \alpha D A^T (b - Ax^k), \tag{4}$$

where α is a gain factor usually set to $\alpha = \frac{1}{\sigma_1^2}$, D is a *shaping matrix* (a polynomial function of $\alpha A^T A$) and x^k is the reconstructed image after the k-th iteration. Here σ_1 is the largest singular value. For convenience, we define the multiplication of A by a vector as a forward projection,

and the multiplication of A^T by a vector as a backward projection. In analyzing the computational efficiency of an iterative method, the number of forward and backward projections usually serves as a better indicator than the number of iterations does [7]. If the order of the polynomial function used to represent D is l, then each single generalized Landweber iteration will need $(l+1)$ forward and backward projections [8]. When initialized with zero, the iteration converges to x^*, provided the Euclidean norm $\|\alpha D A^T A\|_2 < 2$ [2].

One example of D is $D = F(\alpha A^T A)$ where the polynomial function $F(\cdot)$ is chosen to be [2]

$$\begin{aligned} F(\lambda) &= 31.5 - 315\lambda + 1443.75\lambda^2 - 3465\lambda^3 \\ &\quad +4504.5\lambda^4 - 3003\lambda^5 + 804.375\lambda^6. \end{aligned} \quad (5)$$

This choice is made because $\lambda F(\lambda)$ is a good approximation to the unit step function in the range $\lambda \in (0, 1]$. This covers the entire spectrum of frequency components from the highest frequency component $\sigma = 0$ to the lowest frequency component $\sigma = 1$ after the normalization $\alpha = \frac{1}{\sigma_1^2}$.

III. VARIABLE SHAPING

From the recover-and-stay property, we have found that one shaping matrix can be used for the first several iterations to recover the low frequency components of the image, and then another shaping matrix, which requires less computation than the first shaping matrix in a single generalized Landweber iteration, can be used to accelerate the reconstruction of high frequency components. We also found that if the second shaping matrix provides negative gains in a range of high frequency components, the partial recovery of the high frequency components in this range using the first shaping matrix can be removed. The result is very similar to the result using TIF. Two different examples are provided to demonstrate 1) acceleration of the convergence of high frequency components, and 2) attenuation of high frequency components.

A. *Acceleration of Reconstruction of High Frequency Components*

Suppose the shaping matrix $D = F(\alpha A^T A)$, where $F(\cdot)$ is defined in (5), is used in the first three generalized iterations with $\alpha = \frac{1}{\sigma_1^2}$. The frequency components with singular values above 0.2 will be recovered in these three iterations (which are equivalent to 21 forward and backward projections). Fig. 1 shows the filter after the three iterations.

Starting with the fourth iteration, suppose the iteration switches the shaping matrix $D = F(\alpha A^T A)$ to $D' = F'(\alpha A^T A)$, where $F'(\cdot)$ is defined as

$$\begin{aligned} F'(\lambda) &= 25.94897 - 126.47433\lambda \\ &\quad +200.73822\lambda^2 - 100.03043\lambda^3. \end{aligned} \quad (6)$$

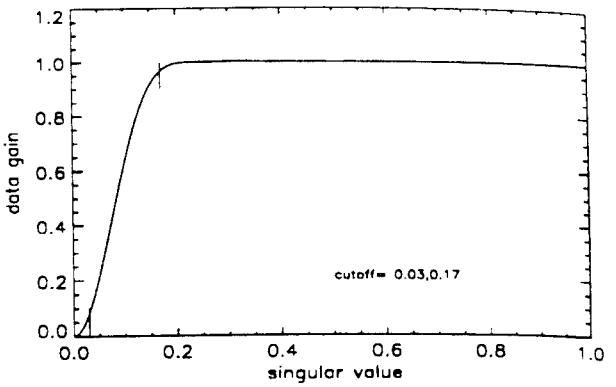


Figure 1: The resulting filter after the twelve iterations.

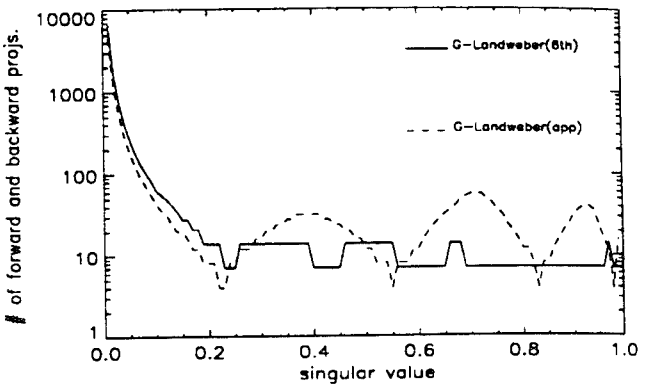


Figure 2: Comparison of performance in recovering 95% of components on various singular values using the generalized Landweber iterations with $F(\lambda)$ (marked as G-Landweber(6th)), and with $F'(\lambda)$ (marked as G-Landweber(app)). It is clear that $F'(\lambda)$ out-performs $F(\lambda)$ in $\sigma \in [0, 0.2]$, since it requires less computation than $F(\lambda)$.

$F'(\lambda)$ is obtained from the Chebyshev approximation [9] of a third-order polynomial to $F(\lambda)$ in (5). The idea is to use a lower order polynomial to approximate $F(\lambda)$ to maintain the response in the range of high frequency components, while relaxing the response in the range of low frequency components, which have already been recovered. Fig. 2 shows that $F'(\lambda)$ actually out-performs $F(\lambda)$ in recovering 95% of the high-frequency components with singular values $\sigma \in (0, 0.2]$.

B. *Attenuation of High Frequencies*

We now demonstrate another way of using a variable shaping matrix: to attenuate high frequency components in the reconstructed image to obtain a smooth image. This can also be viewed as a solution to a regularized image reconstruction problem. Using a shaping matrix $D = F(\alpha A^T A)$ as in (5) will partially recover some high frequency components, which may deteriorate the image quality signifi-

cantly if these components are noisy. Thus it may be desirable to remove the high frequency components, including both the reconstructed image and the reconstructed noise, in the range (say) $\sigma \in (0, 0.1]$, to make the image smooth. We now show how the generalized Landweber iteration can do this *during* the image reconstruction process.

Suppose the polynomial function $F(\lambda)$ in (5) is used for the first three generalized Landweber iterations to ensure that low frequency components with $\sigma \in (0.2, 1]$ have been recovered (see Fig. 1).

Now suppose the following polynomial function [3]

$$\begin{aligned} F''(\lambda) &= -10.972414 + 209.51804\lambda - 1072.02\lambda^2 \\ &\quad -2599.2093\lambda^3 - 3313.8866\lambda^4 \\ &\quad +2146.6546\lambda^5 - 557.55271\lambda^6 \end{aligned} \tag{7}$$

is used for the subsequent iterations. $F''(\lambda)$ provides *negative* gains in the range $\sigma \in (0, 0.28]$, which covers $\sigma \in (0, 0.2]$. $F''(\lambda)$ is used for another nine generalized Landweber iterations to ensure that sufficient attenuation or removal of high frequency components has been achieved.

Now, if $F(\lambda)$ is used for the first three generalized Landweber iterations, and $F''(\lambda)$ is for the subsequent nine iterations, the resulting response after these twelve iterations is similar to the result of a *low-pass filtering* with a *transition band* having low cutoff frequency at $\sigma_{\text{lo}} = 0.19$ and high cutoff frequency at $\sigma_{\text{hi}} = 0.08$. Fig. 3 shows the resulting low-pass filter after the twelve iterations. The transition band cutoff frequencies σ_{lo} and σ_{hi} were chosen at the frequencies with gains 0.95 and 0.05, separately. In general, this type of shaping is also referred to as roll-off shaping [1].

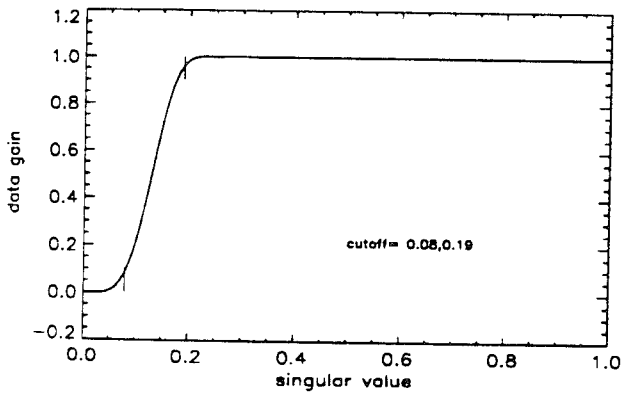


Figure 3: The resulting filter after the twelve iterations.

We used this method to simulate the reconstruction of the complex phantom in the simulated PET-3 geometry from noisy data [3]. The number of total counts was 1 million; the data noise was Poisson in nature. Fig. 4 shows the result of the third and the twelfth iterations. Compare the third and the twelfth reconstructed images of Fig. 4. As expected, the twelfth reconstructed image is smoother than the third image.

An important feature of the generalized Landweber iteration can also be seen in Fig. 4 — not only have we obtained the reconstructed images, but we also have a *known* relation between a reconstructed image and the shaping filter generating the image. This known relation is unique to the generalized Landweber iteration; it is not present in the ART or CG iteration. Since $F(\lambda)$ and $F''(\lambda)$ are both polynomial functions of order 6, $84(= (6+1) \times 12)$ forward and backward projections are *known to be needed* to obtain the desired image *characteristics*.

In examining the images in Fig. 4, note the change due to the reconstruction and removing of the high frequency components in $\sigma \in (0, 0.2]$. The deterioration of the image after the first three iterations was due to the partial reconstruction of these high frequency components.

C. Computational Comparison

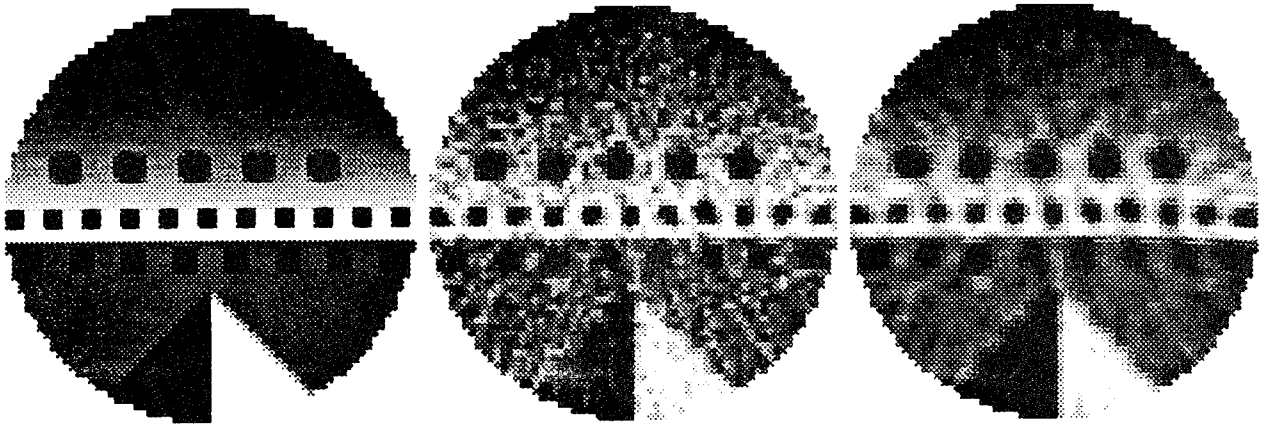
We compare the computational requirements for a roll-off shaping filter (see the previous subsection) and a truncated shaping filter as in TIF. The major difference is that the roll-off shaping can be iteratively performed, while the truncated shaping must be computed directly using an SVD; this requires an enormous amount of computation.

Direct computation of $\boldsymbol{x}^*$ requires storage of the matrices U and V and a considerable amount of computation to implement (3), since U and V (unlike A) are not sparse. It also requires an (off-line) computation of the SVD of A. Consider a 4000×4000 system matrix A, and assume using four bytes to store a number in floating-point format. Then 128 million $(= 4000 \times 4000 \times 4 \times 2)$ bytes will be required just to store U and V! On the other hand, if A is 97% sparse (3% of the elements of A are non-zero), the computation time for obtaining $\boldsymbol{x}^*$ in (3) is equivalent to about 33 $(\approx 2/0.06)$ forward and backward projections since the sparse structure of A can be utilized in the iteration. If k in (3) is chosen to preserve half of the singular values, then 17 forward and backward projections are required for the computation.

In some applications, the system matrix A becomes sparser as its size increases. For example, a 150-million-elements system matrix in [5] has about 2 million non-zero elements, so that it is 98.7% sparse. In this case, an SVD computation will become almost impossible. The computation time for obtaining $\boldsymbol{x}^*$ is equivalent to about $75(\approx 2/0.026)$ forward and backward projections. If half of the singular values are preserved, about 37 forward and backward projections will be needed. Hence *even apart from the storage problem and the difficulty in computing the SVD of* A, the iterative approach of roll-off shaping becomes more favorable over the direct approach using TIF as the size and sparsity of A grows.

IV. Summary

A new variable shaping matrix method for using the generalized Landweber iteration in image reconstruction in



(a) (b) (c)

Figure 4: The reconstructed images from (b) the third and (c) the twelfth iterations. (a) is the complex-phantom image.

emission tomography has been developed. The method uses two shaping matrices: the first one is for fast recovery of low frequency components; and the second one is either for the acceleration of the reconstruction of high frequency components if the noise does not impose a problem in recovering the high frequency components, or for the attenuation of high frequency components when noise is a problem.

Examples of using a variable shaping matrix to accelerate the reconstruction of high frequency components and to attenuate high frequency components were presented. The savings in computation and storage space in using a variable shaping matrix instead of TIF was also presented; it results from the sparsity of A, which generally increases as the size of A increases. Hence the larger the system, the greater the advantage of the new method over TIF.

An important topic for further research is application of the generalized Landweber iteration with a variable shaping matrix to linear-algebra-based signal processing problems. The solution (3) is used in signal restoration problems, harmonic retrieval, and deconvolution. The method proposed in this paper can be applied to all of those problems, even if the matrix is not sparse, the computation required is less than that for a SVD. Another topic is further study on designing the shaping matrix to achieve a given specification of low and high cutoff frequencies with the least amount of computation.

REFERENCES

[1] C. K. Rushforth, "Signal restoration, functional analysis, and Fredholm integral equations of the first kind," in *Image Recovery: Theory and Applications* (H. Stark, ed.), ch. 1, pp. 1–27, Orlando: Academic Press, 1987.

[2] O. N. Strand, "Theory and methods related to the singular-function expansion and Landweber's iteration for integral equations of the first kind," *Siam J. Numer. Anal.*, vol. 11(4), pp. 798–825, Sept. 1974.

[3] T.-S. Pan, *The Generalized Landweber Iteration in Positron Emission Tomography.* The University of Michigan: Dissertation, 1991.

[4] Y. Censor, "Finite series-expansion reconstruction methods," *IEEE Proceedings*, vol. 71(3), pp. 409–419, Mar. 1983.

[5] L. Kaufman, "Implementing and accelerating the EM algorithm for positron emission tomography," *IEEE Trans. Medical Imaging*, vol. MI-6(1), pp. 37–51, Mar. 1987.

[6] T. F. Budinger, G. T. Gullberg, and R. H. Huesman, "Emission computed tomography," in *Reconstruction from Projections: Implementation and Applications* (G. T. Herman, ed.), ch. 5, pp. 147–246, New York: Springer-Verlag, 1979.

[7] T.-S. Pan and A. E. Yagle, "Acceleration of Landweber-type algorithms by suppression of projection on the maximum singular vector," *submitted to IEEE Trans. Med. Imaging*, 1991.

[8] T.-S. Pan and A. E. Yagle, "Numerical study of multigrid implementations of some iterative image reconstruction algorithms," *to appear on IEEE Trans. Med. Imaging*, Dec. 1991.

[9] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes in C.* New York: Cambridge University Press, 1988.

APPENDIX G1

T.-S. Pan and A.E. Yagle, "Acceleration of Landweber-Type Algorithms by Suppression of Projection on the Maximum Singular Vector," IEEE Trans. Medical Imaging 11(4), 479-487, December 1992.

This paper presents a simple procedure to significantly accelerate the convergence of the generalized Landweber iteration.

Acceleration of Landweber-Type Algorithms by Suppression of Projection on the Maximum Singular Vector

Tin-Su Pan, *Member, IEEE*, and Andrew E. Yagle, *Member, IEEE*

Abstract— **We develop a new procedure that speeds up convergence during the initial stage (the first 100 forward and backward projections) of Landweber-type algorithms, iterative image reconstruction for PET, which include the Landweber, generalized Landweber, and steepest descent algorithms. The procedure first identifies the singular vector associated with the maximum singular value of the PET system matrix, and then suppresses projection of the data on this singular vector after a single Landweber iteration. We show that typical PET system matrices have a significant gap between their two largest singular values; hence, this suppression allows larger gains in subsequent iterations, speeding up convergence by roughly a factor of three. New contributions of this paper include: 1) study of the singular value spectra of typical PET system matrices, 2) study of the effect on convergence of projection on the maximum singular vector, and 3) study of the convergence behavior of the new procedure applied to the Landweber, generalized Landweber, steepest descent, conjugate gradient, and ART algorithms (comparison is also made with the MLEM algorithm).**

I. Introduction

RECENTLY, there has been growing interest in using iterative reconstruction algorithms in tomographic imaging [1]–[7]. In using an iterative reconstruction algorithm, the presence of noise from the imaging process and roundoff error requires imposing a stopping rule to stop the iteration before the statistical noise in the reconstructed image becomes too big [3]–[10], or incorporating a regularization strategy into the algorithm [11]–[13].

Most research has focused on the following iterative reconstruction algorithms: algebraic reconstruction technique (ART) [14], [15], maximum-likelihood-expectation-maximization (MLEM) iteration [16], steepest descent (STP) algorithm [13], [17], and conjugate gradient (CG) algorithm [13], [17]. A problem with these algorithms is that there is no clear formula to describe what has been achieved after some number of iterations. Each ART iteration modifies the reconstructed image by projecting from one hyperplane to another hyperplane defined by a system of linear equations; MLEM iteration maximizes a likelihood function; and each STP and CG iteration searches adaptively for the largest gradient defined by the least-square error between the projection data and the estimated projection data. In each case, the precise meaning of the image following each iteration is unclear.

A solution to this problem is to use a more controllable algorithm for which the reconstruction process can be examined analytically. Using the Landweber [18] or the generalized Landweber iteration [19], [20], one knows exactly what singular value spectral components are recovered in each iteration. The generalized Landweber iteration, with a proper shaping matrix, can accelerate and regularize the reconstruction process [20]; the number of iterations needed and the convergence behavior can be designed before the iteration begins, and computation time can be estimated accurately.

In this paper we propose a new method, which we call the *DC-suppression procedure*, for improving the convergence rate of the Landweber-type iterations, which include the Landweber, generalized Landweber, and STP algorithms. The "DC component" of an image is defined here as its component along the singular vector associated with the system matrix's maximum singular value. We first demonstrate numerically that typical PET system matrices have a significant gap between their two largest singular values (a rigorous proof was provided in Johnstone and Silverman [21]). Also, the DC component of an image can be completely recovered by the first Landweber iteration. By suppressing this component in subsequent iterations, the gain factor in the iteration can be increased, speeding up the convergence of the other image components. For practical applications, we will only investigate the reconstruction speed of various algorithms in the first 100 forward and backward projections.

II. System Equation

The system equation [15], [16] that describes the transformation or projection process in tomographic imaging (e.g., SPECT or PET) is usually represented as

$$Ax = b \tag{1}$$

where A is an $m \times n$ system matrix, which describes the system geometry, x is an $n \times 1$ vector of the image pixels, and b is an

Manuscript received April 30, 1991; revised March 12, 1992. The work of T.-S. Pan was supported in part by the NIH under Grant PO1–CA42768 and in part by a Research Partnership award from the H. H. Rackham School of Graduate Studies of the University of Michigan. The work of A. E. Yagle was supported in part by the ONR under Grant N00014-90-J-1897.

T.-S. Pan was with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109. He is now with the Department of Nuclear Medicine, University of Massachusetts Medical Center, Worcester, MA 01655.

A. E. Yagle is with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109.

IEEE Log Number 9201734.

0278–0062/92$03.00 © 1992 IEEE

$m \times 1$ vector of the measured projections of the image. The matrix A is typically very sparse [5].

There are several problems associated with the solution of (1):

1) the size of the system—typically m and n are on the order of several thousand [5], [16];

2) ill-conditioning of the system—typically A has a large condition number [22], so that a small change in the projection data b may cause a large change in the solution x or in the minimum-norm least-squares solution x^*,

$$x^* = A^\dagger b \tag{2}$$

where $A^\dagger$ is the pseudoinverse of A (see (3) below) [23];

3) Ill-posedness—typically A may have a nonempty null space, so that some components of the image cannot be recovered from the projection b without additional information.

The notations in singular value decomposition (SVD) [20], [23] are adopted here for the convenience of discussion. The SVD of A is $A = U\Sigma V^T$. Here $U = [u_1, u_2, \cdots, u_m]$ and $V = [v_1, v_2, \cdots, v_n]$ are orthogonal matrices, and Σ is "diagonal" in that $\langle\Sigma\rangle_{i,j} = 0$ unless $i = j$, in which case $\langle\Sigma\rangle_{i,i} = \sigma_i$, the singular values of A. Without loss of generality, let $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{\rho(A)} > 0$, where $\rho(A)$ is the rank of A. It can be shown [23] that the minimum-norm least-squares solution to (1) is

$$x^* = V\mathrm{diag}\left[\frac{1}{\sigma_1} \cdots \frac{1}{\sigma_{\rho(A)}} 0 \cdots 0\right] U^T b = \sum_{i=1}^{\rho(A)} \frac{1}{\sigma_i}(b, u_i)v_i \tag{3}$$

where $(b, u_i) = b^T u_i$ is the inner product of vectors b and u_i.

III. Landweber-Type Reconstruction Algorithms

Throughout this section, we assume that A has maximum singular value $\sigma_1 = 1$. We relax this assumption in Section IV. While some of this material is taken from [18], [19], and [20], we make some important new points at the end.

A. Definition

We define a *Landweber-type iteration* as an iteration of the form

$$x^{k+1} = x^k + \alpha_k DA^T(b - Ax^k) \tag{4}$$

where α_k is a scalar and x^k is the reconstructed image after the kth iteration. Equation (4) becomes the following:

1) the *Landweber iteration* [18] when $\alpha_k = \alpha$ (a constant) and $D = I$ (identity matrix):

$$x^{k+1} = x^k + \alpha A^T(b - Ax^k); \tag{5}$$

2) the *generalized Landweber iteration* [19] when $\alpha_k = \alpha$ (a constant) and D is a shaping matrix (a polynomial function of $\alpha A^T A$, see (11) and (12) below):

$$x^{k+1} = x^k + \alpha DA^T(b - Ax^k); \tag{6}$$

3) the *STP iteration* when $D = I$ and α_k adaptively changes at each iteration:

$$x^{k+1} = x^k + \alpha_k A^T(b - Ax^k) = x^k + \alpha_k q_k,$$
$$q_k = A^T(b - Ax^k), \quad \alpha_k = \frac{q_k^T q_k}{(Aq_k)^T Aq_k}. \tag{7}$$

For simplicity, we will refer to both α and α_k as gain factors.

The results of [18], [19], and [20] summarized below all assume that the maximum singular value of A is $\sigma_1 = 1$; this is why we also make that assumption in this section. In Section IV, we relax this assumption; this is why we include the gain α [which scales $A^T A$ in (5) and (6)]. For the rest of this section, assume $\alpha = 1$ unless stated otherwise; we continue to exhibit α for later convenience.

When initialized with zero, all three Landweber-type iterations will converge to the minimum-norm least-squares solution (3) [17], [20], provided $\|\alpha A^T A\|_2 < 2$ [18], [24] (for the Landweber iteration) or $\|\alpha DA^T A\|_2 < 2$ [19], [25] (for the generalized Landweber iteration).

B. Convergence and Filtering Control of the Generalized Landweber Iteration

Define the *components* of an image as its projections on the singular vectors v_i. These are analogous to (but not the same as) Fourier frequency components, with "high frequencies" associated with small singular values σ_i, and "low frequencies" associated with large singular values [20]. Similar arguments were adopted also in Barret *et al.* [26] and Smith *et al.* [22]. This terminology is common (e.g., [12]), although these frequencies coincide with Fourier frequencies only in a special circumstance [10]. In the sequel, the image component on the singular vector associated with σ_1 is referred to as the "DC component," since it has the smallest "frequency."

Let the initial condition be zero. In the generalized Landweber iteration, the definition of D will be complete if the image Dv_i of each singular vector v_i in $R(A^T)$ (range space of A^T) is specified [19], [20]. Strand [19] proposed that the matrix D could be designed by specifying the scalars $p_1, \cdots, p_{\rho(A)}$ in

$$Dv_i = p_i v_i, \quad 0 < p_i\sigma_i^2 < 2, \tag{8}$$

where the condition $0 < p_i\sigma_i^2 < 2$ ensures the convergence of the generalized Landweber iteration. The reconstructed image x^k after the kth generalized Landweber iteration can be represented as [20]

$$x^k = \sum_{i=1}^{\rho(A)} \left[1 - \left(1 - p_i\sigma_i^2\right)^k\right] \frac{1}{\sigma_i}(b, u_i)v_i. \tag{9}$$

Note that the gain after k iterations (compare (3) and (9))

$$G(\sigma_i, k) = \left[1 - \left(1 - p_i\sigma_i^2\right)^k\right], \tag{10}$$

illustrates the control over both convergence and filtering that is possible in the generalized Landweber iteration. By varying the p_i's, it is possible to control the convergence rates of each component of the image independently. And by varying k, the extent of convergence can be controlled—some components

of the image can be partially filtered out by stopping the iteration early. The latter control is also available in the Landweber iteration.

C. Examples—Three Choices for D

We now demonstrate the convergence behavior of the generalized Landweber iteration for three different choices of the shaping matrix D. First, choose $D = I$, so that $p_i = 1$ and the generalized Landweber iteration reduces to the Landweber iteration. Equation (10) makes it clear that high-frequency components of the image (those associated with small σ_i) converge more slowly than low-frequency components (those associated with large σ_i). Indeed, since $\sigma_1 = 1$, *the DC component converges* (i.e., *is completely recovered*) *after a single Landweber iteration*. This fact plays a vital role below.

Second, let D be specified by choosing $p_i = \frac{1}{\sigma_i^2}, \sigma_i \neq 0$ in (9). Then the reconstructed image after the first iteration is identical to the minimum-norm least-squares solution (3)! However, this choice is not practical, since a lengthy SVD computation of A would be needed to obtain the p_i. The third choice of D is from [19], in which D is given as a polynomial function $F(\cdot)$ of $\alpha A^T A$, i.e., $D = F(\alpha A^T A)$ where

$$F(\lambda) = 31.5 - 315\lambda + 1443.75\lambda^2 - 3465\lambda^3 + 4504.5\lambda^4 - 3003\lambda^5 + 804.375\lambda^6. \quad (11)$$

The polynomial function $F(\lambda)$ is chosen so that $\lambda F(\lambda)$ is a good approximation to the unit step function in the range $\lambda \in [0, 1]$. This choice results in (see (8); assume $\alpha = 1$)

$$Dv_i = F(A^T A)v_i = F(\sigma_i^2)v_i = p_i v_i \quad (12)$$

implying $F(\sigma_i^2) = p_i$, so that (11) effectively chooses $p_i \sigma_i^2 \approx 1$ to achieve the minimum-norm least-squares solution faster [see (9)]. In the sequel, we make this choice of D in the generalized Landweber iteration.

These convergence properties of the Landweber and the generalized Landweber iterations are illustrated in Fig. 1(a) and (b), respectively. It is clear for *both* algorithms that the low-frequency components (larger singular values) converge faster than the high-frequency components (smaller singular values)—the farther a component's singular value is from one, the slower it converges. This is called the nonuniform convergence property in [20].

D. Effect of DC Component on Convergence

Although the DC component converges after a single Landweber iteration, Fig. 1(a) makes it clear that other components in the Landweber iteration converge more slowly. Furthermore, the smaller σ_i is (relative to $\sigma_1 = 1$), the slower the image component on v_i will converge. For the generalized Landweber iteration, Fig. 1(b) shows that the low-frequency components (associated with $\sigma_i > 0.2$) all converge quickly, but the high-frequency components (associated with $\sigma_i < 0.2$) converge more slowly.

Since the DC component converges immediately, it might seem tempting to make $\alpha > 1$ in (5) and (6). Since $D = F(\alpha A^T A)$, this scales all of the singular values of A by $\sqrt{\alpha}$, bringing the smaller ones closer to one, so that their associated

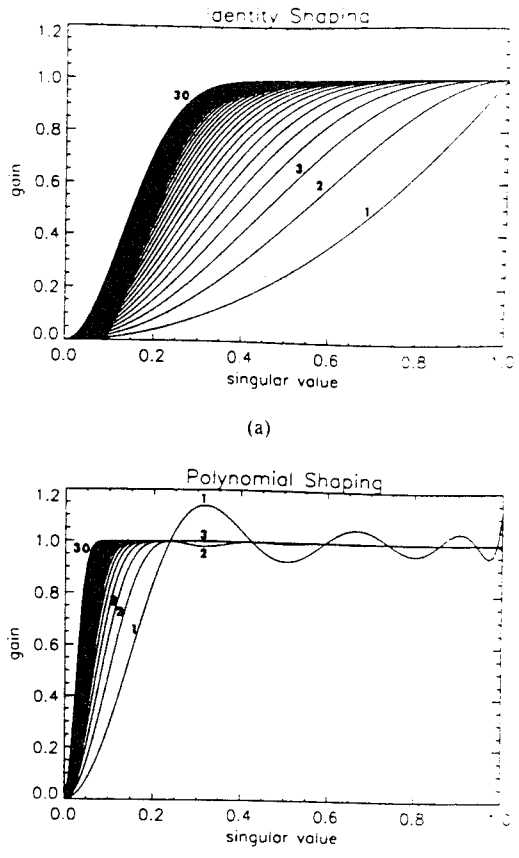


Fig. 1. The convergence properties of (a) the Landweber and (b) the generalized Landweber iterations. These curves were derived from $G(\sigma, k)$, with k ranging from 1 to 30, for the Landweber iteration ($p_i = 1$) and the generalized Landweber iteration ($p_i = F(\sigma_i^2)$). The first, second, and third iteration plus the last iteration is labeled.

components converge faster. Since the condition for stability in the generalized Landweber iteration is $\|\alpha D A^T A\|_2 < 2$, there seems to be a good margin. However, $D = F(\alpha A^T A)$ is a matrix polynomial of degree p, each term having eigenvectors v_i, and assuming that the highest-degree term dominates results in

$$\|\alpha D A^T A\| \approx \alpha^{p+1} \|A^T A\| = \alpha^{p+1} < 2. \quad (13)$$

Hence, if $\alpha > \sqrt[p+1]{2}$ then the generalized Landweber iteration will diverge. For the Landweber iteration ($p = 0$), $\alpha > 2$ makes the iteration diverge. In particular, $p = 6$, if the polynomial function (11) is used, implies that $\alpha > \sqrt[7]{2} = 1.104$ will make the generalized Landweber iteration diverge; hence increasing α significantly is not a viable option. Indeed, due to roundoff error, it is prudent to bound α by $0 < \alpha \leq 1$ in the generalized Landweber algorithm. In the Landweber algorithm, α may be as large as 2 without endangering stability, but increasing α will slow down the convergence of low-frequency components [let $p_i = 1$ and see (10)]. Therefore, α is usually set equal to one [19].

To see precisely what happens when $\alpha > 1$, recall from (12)

$$\alpha D A^T A v_i = F(\alpha A^T A)(\alpha A^T A)v_i = (\alpha\sigma_i^2)F(\alpha\sigma_i^2)v_i = \lambda F(\lambda)v_i, \quad \lambda = \alpha\sigma_i^2. \quad (14)$$

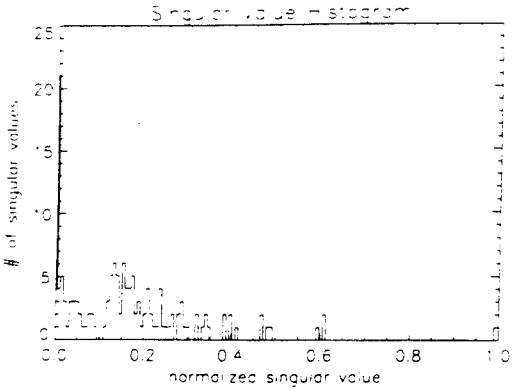


Fig. 2. The singular value histogram of the system matrix A_1. Note the singular values have been normalized or divided by their largest singular value.

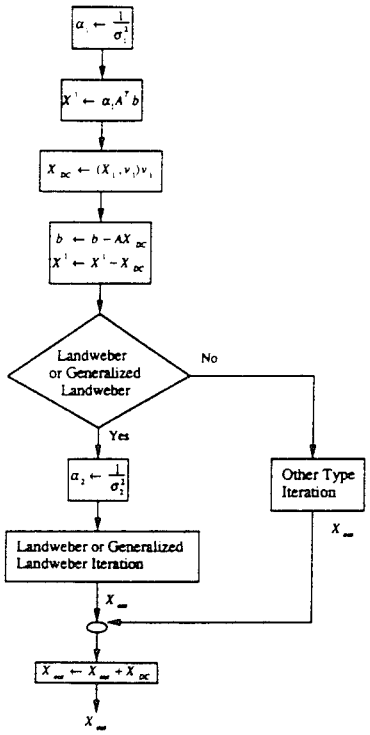


Fig. 3. The flowchart of the DC-suppression procedure.

But $\lambda F(\lambda) \approx 1$ only for $\lambda \in [0, 1]$; outside this interval $\lambda F(\lambda) \notin [0, 2]$ and the algorithm diverges. Hence stability requires that $\lambda = \alpha\sigma_i^2 \leq 1$, so that $\alpha \leq \frac{1}{\sigma_i^2}$ for all i, which requires $\alpha \leq \frac{1}{\sigma_1^2}$.

IV. The New DC-Suppression Procedure

We now relax the assumptions that the maximum singular value σ_1 of A is one, and that $\alpha = 1$. It is clear that choosing $\alpha = \frac{1}{\sigma_1^2}$ reduces this general case to the case where the maximum singular value of A is one, since $D = F(\alpha A^T A)$ and α multiplies $A^T A$ in (5) and (6). Then all of the results of Section III apply.

A. Motivation

Three elements motivated the development of the new DC-suppression procedure. First, the gain factor α is effectively bounded by $\frac{1}{\sigma_1^2}$ (see above), and Fig. 1(a) and (10) show that the DC component of the image, associated with σ_1, is recovered after a single Landweber iteration. This DC component impedes the speed of reconstruction of the other components, by limiting the gain factor α. However, if the DC component were removed from the iteration, the gain factor α could be increased to $\frac{1}{\sigma_2^2}$ without causing instability. This would speed up the reconstruction of the other frequency components, since their associated singular values would be scaled closer to one.

Second, the system matrix A for PET systems has all non-negative elements, as does $A^T A$. Such matrices tend to have a large gap between their two largest singular values σ_1 and σ_2; bounds on the ratio $r_\sigma = \frac{\sigma_1}{\sigma_2}$ have been given in [27]. Hence we expect σ_1 to be significantly larger than $\sigma_2, \sigma_3, \cdots$. This phenomenon was observed in analyzing the distribution of the singular values of the system matrix A_1 in [20]. A_1 was the 195×144 system matrix describing a synthetic PET geometry with 12×12 pixels centered in a detector ring of radius 1, which hosted 26 identical detectors around the ring circle. Fig. 2 shows the histogram of the singular values of A_1; there is a large gap between the two largest singular values ($\sigma_1 = 1.0433, \sigma_2 = 0.61995$, the ratio $r_\sigma = \frac{\sigma_1}{\sigma_2} = 1.683$). The singular-value spectra of larger system matrices also exhibits this phenomenon; see Section V. Since the singular values except σ_1 are close together (see Fig. 2, and compare with the gap between σ_1 and σ_2), to derive σ_3 and v_3 will not be easy; there is not much one can gain from deriving σ_3 and v_3.

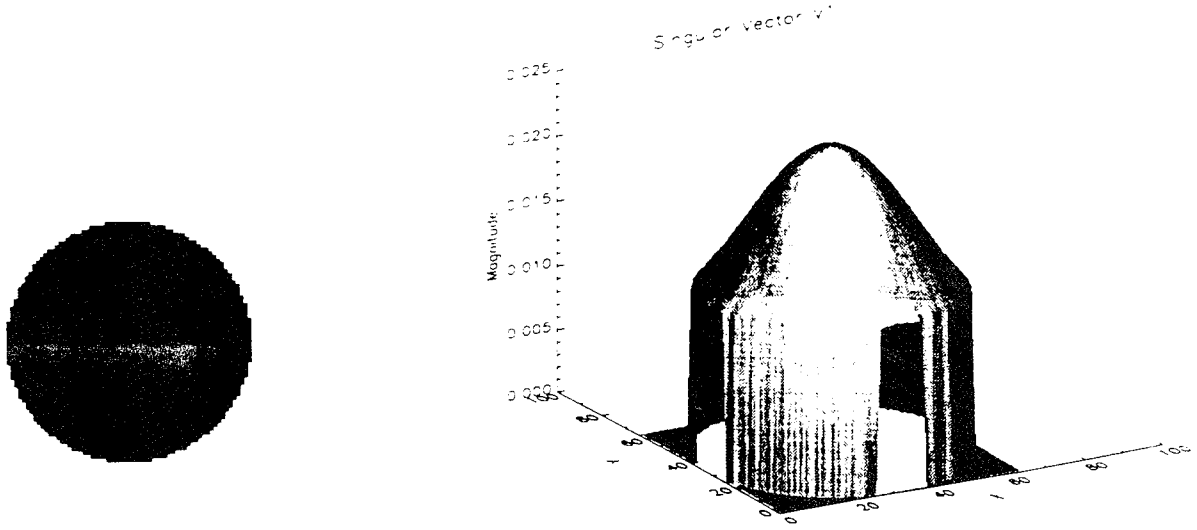
Finally, to remove the recovered DC component from the reconstructed image x^1 after the first Landweber iteration, we need: 1) the maximum singular value σ_1 of the system matrix A, to set $\alpha_1 = \frac{1}{\sigma_1^2}$; 2) the singular vector v_1, so that the projection $(x^1, v_1)v_1$ of x^1 on v_1 can be removed from subsequent Landweber-type iterations; and 3) the second-largest singular value σ_2, so that $\alpha_2 = \frac{1}{\sigma_2^2} > \alpha_1$ can be used in the subsequent Landweber or generalized Landweber iterations, speeding up the convergence of the other image components by scaling their associated singular values closer to one. Note that the STP iteration does not use α_2.

B. Determination of σ_1, v_1, and σ_2: The Power Method

The *power method* [28] is used to calculate σ_1 and v_1. After determining v_1, the method with a *purification* procedure [28] can be used to determine σ_2 and v_2. The power method is an iterative method; convergence of the iteration for deriving σ_1 and v_1 depends on the ratio $r_\sigma = \frac{\sigma_1}{\sigma_2}$—the larger r_σ is, the quicker the convergence will be. Since r_σ is relatively large for PET system matrices, the power method converges relatively quickly. The initial conditions for finding v_1 and v_2 are $[1, 1, \cdots, 1]$ (all 1's) and $[1, 1, \cdots, 1, -1, -1, \cdots, -1]^T$, (half 1's and half -1's), respectively, since they are geometrically similar to the DC and first harmonic components in a LSI system.

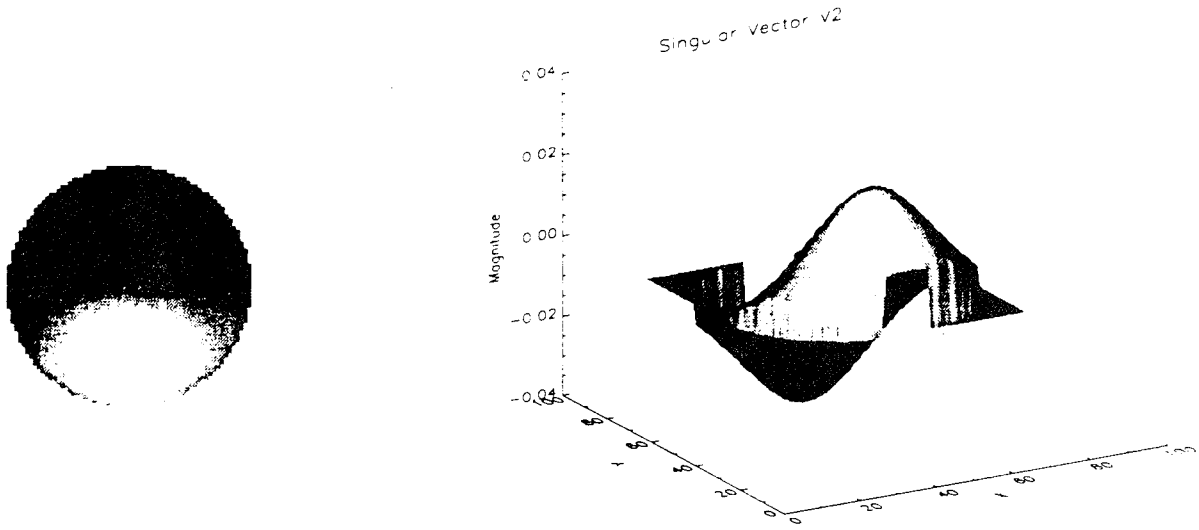
C. The New DC-Suppression Procedure

The new DC-suppression procedure is as follows.



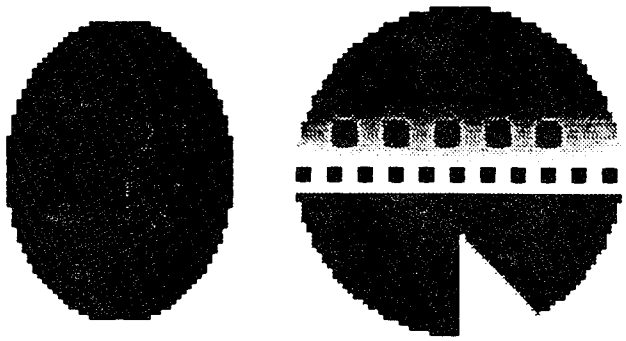
(a) (b)

Fig. 4. (a) The singular vector v_1 and (b) its 3-D shade plot. Note the openings in (b) are only for 3-D display purpose; the real image does not have these openings.



(a) (b)

Fig. 5. (a) The singular vector v_2 and (b) its 3-D shade plot. Note the openings in (b) are only for 3-D display purpose; the image does not have these openings.



(a) (b)

Fig. 6. (a) The Shepp–Logan phantom and (b) the complex phantom.

1) Compute σ_1, v_1, and σ_2 using the power method.
2) Set $\alpha_1 = \frac{1}{\sigma_1^2}$ and perform a *single Landweber iteration*, yielding x^1.
3) Compute the *DC component* of the image $x_{\mathrm{DC}} = (x^1, v_1)v_1$.
4) *Suppress* the DC component of the image by subtracting x_{DC} from x^1.
5) Set $\alpha_2 = \frac{1}{\sigma_2^2}$ for the subsequent Landweber or generalized Landweber iterations.
6) At each subsequent Landweber or generalized Landweber iteration, *suppress* the DC component of the image x^k by subtracting $(x^k, v_1)v_1$ from x^k.
7) After termination, add x_{DC} to $x^{k_{\mathrm{final}}}$.

Step 6) is necessary due to computational roundoff error—theoretically there should be no component of x^k on v_1, but roundoff error will create a very small component in the (generalized) Landweber iteration, which accumulates as the

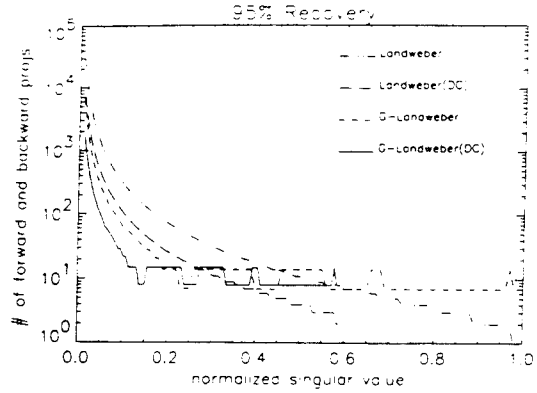


Fig. 7. The number of forward and backward projections needed for the recovery of 95% of components on various singular vectors. Without loss of generality, the largest singular value is assumed to be 1. The symbol DC after Landweber means Landweber with DC suppression; the G-Landweber means generalized Landweber. Note that $r_\sigma = 1.68$.

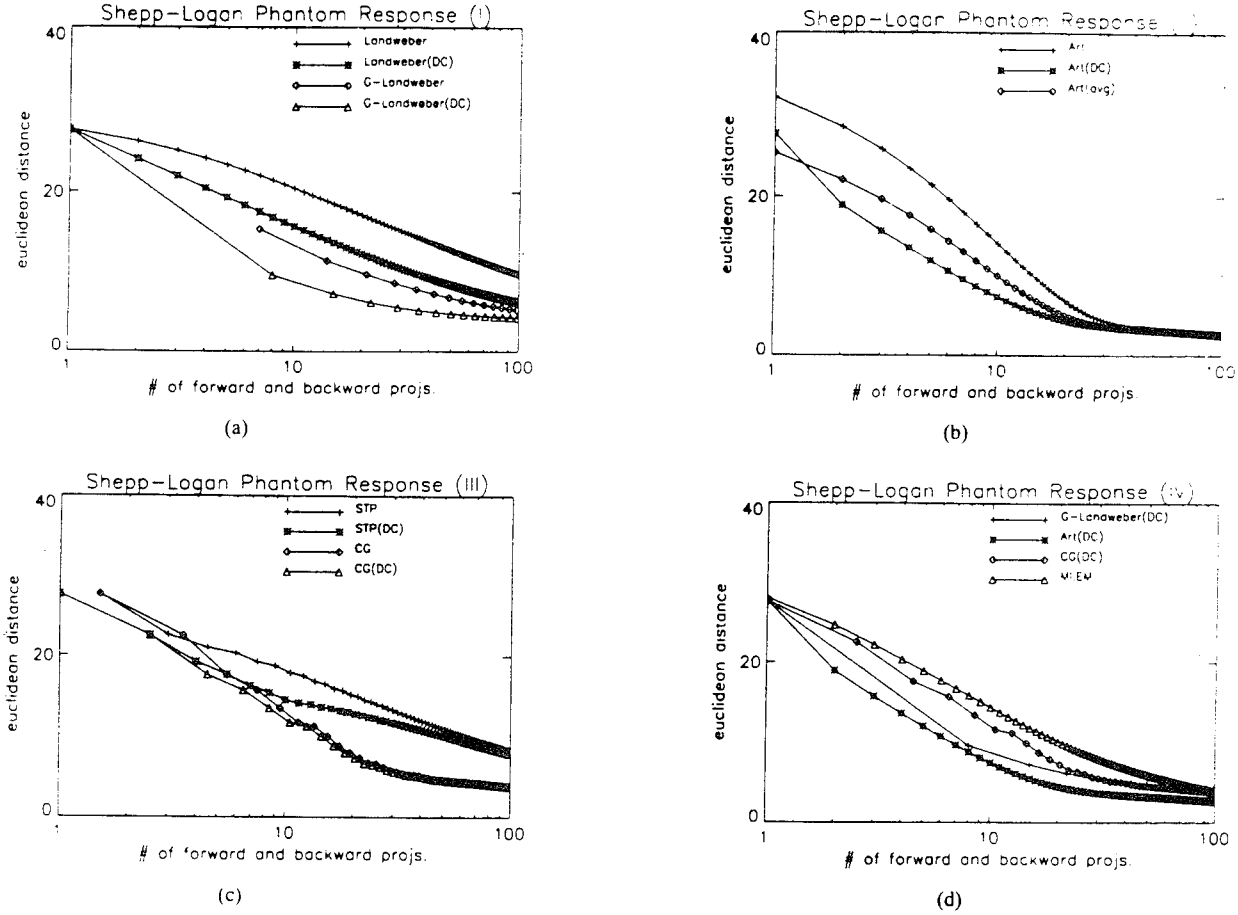


Fig. 8. The performance of reconstructing the Shepp–Logan phantom using (a) the Landweber iteration, the generalized Landweber iteration, and their DC suppression implementations, (b) ART iterations with zero, average, and DC suppression implementation, (c) STP iteration, CG iteration, and their DC suppression implementations, and (d) the fastest methods in the previous three figures (one from each figure) and MLEM iteration.

iteration proceeds. The DC-suppression procedure is illustrated by a flowchart in Fig. 3.

V. Numerical Experiments and Discussions

A. Data Preparation

The PET geometry in [29] (similar to that of [16]), with 128 detectors and a square array of $4096 (= 64 \times 64)$ pixels circumscribed by the detector ring, was used. The object support was assumed to be in the inscribed circle of the square array, accounting for 3228 pixels (determined by counting center positions of pixels inside the circle). A tube is defined by any two different detectors [16]; some tubes defined in this way may not intersect any pixel of the image at all, and these were omitted to reduce computation time. Theoretically, there were $8128 = (128 \times 127/2)$ different tubes for this geometry; after omitting tubes, this was reduced to 4160 tubes. Hence the corresponding system matrix A_2 was 4160×3228.

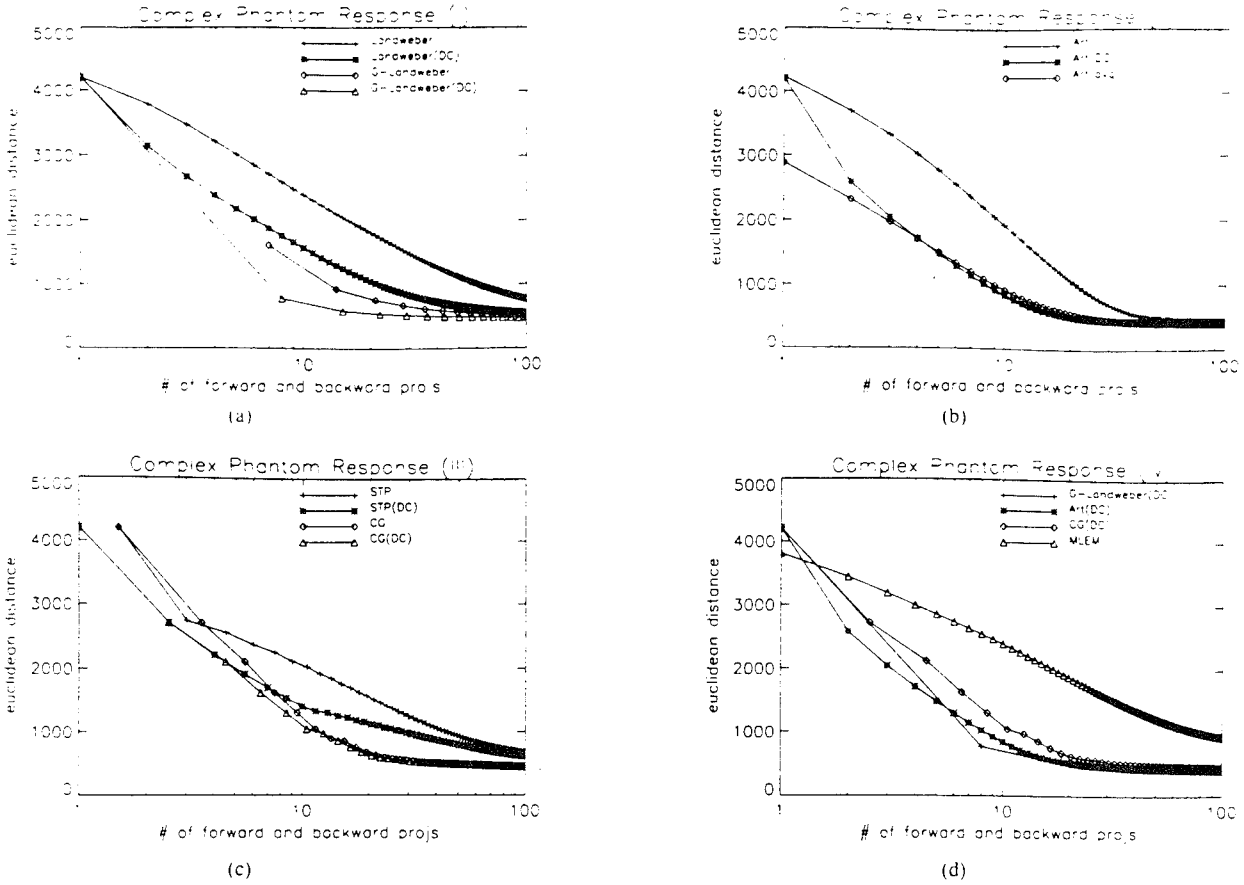


Fig. 9. The performance of reconstructing the complex phantom using (a) the Landweber iteration, the generalized Landweber iteration, and their DC suppression implementations, (b) ART iterations with zero, average, and DC suppression implementation, (c) STP iteration, CG iteration, and their DC suppression implementations, and (d) the fastest methods in the previous three figures (one from each figure) and MLEM iteration.

We assume the conditional probability $P(t/i)$ of an annihilation occurring at pixel i and its consequent coincidence event being detected by tube t is proportional to the joint angle of pixel i to tube t, which is

$$P(t/i) = \frac{\text{joint angle in radians from pixel } i \text{ to tube } t}{2\pi}. \tag{15}$$

A normalization [16] makes $\sum_{t=1}^{t^*} P(t/i) = 1$ where t^* is the total number of tubes intersecting pixel i. The system matrix A_2 is defined by having $(A_2)_{t,i} = P(t/i)$ after normalization.

The singular vectors v_1 and v_2 of A_2 are shown in Figs. 4 and 5, respectively. v_1 looks roughly like the DC component in a LSI system; v_2 looks like the first harmonic in a LSI system. v_1 and $\sigma_1 = 0.986703$ were computed using 9 iterations of the power method; v_2 and $\sigma_2 = 0.587256$ were computed using 25 iterations of the power method with the purification procedure. The ratio $r_\sigma = 1.680$ was strikingly close to the ratio $r_\sigma = 1.683$ of A_1 (dimension 195×144), due to similarity between the geometries.

If one considers the attenuation correction for the object, the system matrix must be modified each time when imaging a different object. However, the computation using the power method to derive the singular values and singular vectors can be carried out in parallel with the data acquisition process. So the overhead in computation can be minimized.

The Shepp–Logan phantom [30] and the complex phantom of [31] shown in Fig. 6(a) and (b), respectively, were used as the test objects or images. Using two different objects avoided bias in interpreting the numerical results.

B. Acceleration of Convergence in the Landweber and the Generalized Landweber Iterations for $r_\sigma = 1.68$

To show how the DC-suppression procedure accelerates convergence, let $r_\sigma = 1.68$, which approximates the values of r_σ for the system matrices A_1 and A_2. Define multiplication of A by an image vector (of dimension n) as a *forward* projection, and multiplication of A^T by a projection data vector (of dimension m) as a *backward* projection. Most computation in iterative reconstruction algorithms consists of these two operations.

The computation for DC suppression consists of one forward and backward projection (see the first four blocks in Fig. 3). A single Landweber iteration also requires one forward and backward projection. The generalized Landweber iteration, with $D = F(\alpha A^T A)$ and $F(\cdot)$ a polynomial of degree p, requires $p+1$ forward and backward projections [20]. In the examples to follow, $F(\cdot)$ is given in (11) and $p = 6$.

Fig. 7 shows the number of forward and backward projections needed to recover 95% of image components on various singular vectors when $r_\sigma = 1.68$. Without loss of generality, the largest singular value is assumed to be 1; the curves

were derived from $G(\sigma, k)$. Fig. 7 shows that after about 22 forward and backward projections (1 from the DC suppression and 21 from 3 generalized Landweber iterations): 1) the generalized Landweber iteration recovers more components than the Landweber iteration, 2) the (generalized) Landweber iteration with DC suppression recovers more high-frequency components than the (generalized) Landweber iteration alone, and 3) the generalized Landweber iteration with DC suppression recovers all image components over 95% on singular vectors with singular values greater than 0.1; to achieve this without DC suppression, 63 forward and backward projections are necessary.

Hence the Landweber and generalized Landweber iterations without DC suppression require roughly three times as many projections to reach the same solution as the iterations with DC suppression. This factor-of-three speed-up, together with the possibility of parallel implementation, suggests greater application of these algorithms in the future.

C. Numerical Experiments

The following iterative image reconstruction algorithms were considered: 1) MLEM [16], 2) ART [15], 3) STP [13], 4) CG [13], 5) Landweber [18], and 6) generalized Landweber [19]. Computation load for each algorithm was represented as the number of forward and backward projections, as follows:

1) each MLEM, ART, or Landweber iteration requires 1 forward and 1 backward projections,
2) each STP iteration requires 2 forward and 1 backward projections,
3) the first CG iteration requires 2 forward and 1 backward projections; each succeeding CG iteration requires 2 forward and 2 backward projections, and
4) each generalized Landweber iteration requires 7 forward and 7 backward projections.

In addition, the DC-suppression procedure requires an additional forward and backward projection.

Figs. 8 and 9 show the numerical results for reconstructing the Shepp–Logan phantom and the complex phantom, respectively. The vertical axis represents euclidean distance between the original image and the reconstructed image; the horizontal axis represents the number of forward and backward projections. The iterations were initialized with zero, unless otherwise specified. We discovered that ART with average initial condition [20] (every pixel is set equal to the number of total counts divided by the number of pixels) is comparable to ART with DC suppression; this was therefore included for comparison. An average initial condition was also used in MLEM, but DC suppression cannot be applied to MLEM since a reconstructed image will then have negative pixels. Results for the first 100 forward and backward projections were compared.

D. Discussion of Results

Figs. 8 and 9 illustrate several points.

1) The ART, Landweber, generalized Landweber, and STP iterations all have their convergence rates accelerated by DC

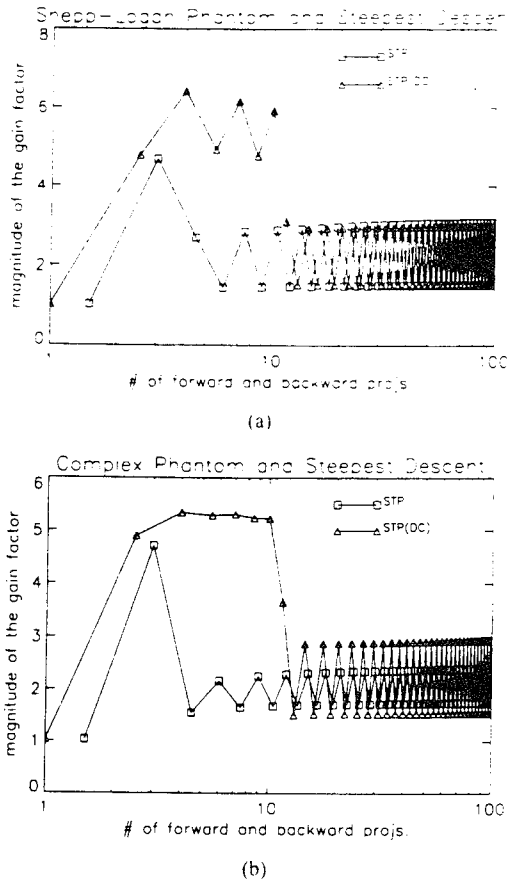


Fig. 10. The change of the gain factors in STP iteration after DC suppression for the reconstruction of (a) the Shepp–Logan phantom and (b) the complex phantom.

suppression; CG does not have its convergence accelerated; and DC suppression cannot be used for MLEM.

2) The generalized Landweber iteration with DC suppression was comparable to ART with DC suppression for the complex phantom [see Fig. 9(d)], but it was a little slower than ART for the Shepp–Logan phantom [see Fig. 8(d)]. Both ART and the generalized Landweber were faster than CG with DC suppression, and MLEM (for about the first 40 forward and backward projections). Although CG will converge to the solution in a finite number of iterations [32], it is not faster during these first 40 forward and backward projections.

3) ART with average initial condition was comparable to ART with DC suppression [see Fig. 8(b) and Fig. 9(b)]. This indicates that by setting the initial condition close to the solution, ART can be significantly accelerated. However, an average initial condition does not accelerate the convergence rate of the (generalized) Landweber iteration [20].

ART and the generalized Landweber iteration, with DC suppression, perform similarly in terms of convergence behavior, measured vs. number of projections. However, the generalized Landweber iteration can be computed in parallel, while ART, a ray-by-ray iteration [15], is not. And the generalized Landweber algorithm allows considerable control over its convergence behavior, unlike ART.

Also note that the acceleration of STP with DC suppression was directly related to the large gain factors α_k in the first several iterations, caused by DC suppression. Fig. 10 shows

this phenomenon: STP with DC suppression had larger gain factors than STP iteration without it for the first 10 forward and backward projections for the Shepp–Logan phantom, and for the first 12 for the complex phantom.

VI. Summary

A new DC-suppression procedure has been developed, and shown to significantly accelerate convergence of the Landweber-type iterations. The procedure identifies and removes the DC component from the iteration, permitting larger gains α_k and accelerating convergence.

The generalized Landweber iteration with DC suppression was comparable to ART with DC suppression. However, considering the feasibility of parallel computing of the forward and the backward projections, and the controllability of the generalized Landweber iteration, the generalized Landweber iteration may be favorably compared to ART.

Topics for further research include further study of the ratio r_σ for different system matrices, and investigation of the possibility of using different shaping matrices to speed up the generalized Landweber iteration with DC suppression.

Acknowledgment

The authors wish to thank Dr. W.L. Rogers and N. Clinthorne of the Nuclear Medicine Division of the University of Michigan Medical Center for many helpful discussions.

References

[1] W.L. Rogers and N.H. Clinthorne, "Fundamental elements of single photon emission computerized tomography," in *Radiology: Diagnosis-Imaging-Intervention*, vol. 1, Philadelphia: Lippincott, 1988, ch. 23, pp. 1–8.

[2] E. Tanaka, "Recent progress on single photon and positron emission tomography—From detectors to algorithms," *IEEE Trans. Nuc. Sci.*, vol. NS-34, pp. 313–320, Feb. 1987.

[3] ——, "A fast reconstruction algorithm for stationary positron emission tomography based on a modified EM algorithm," *IEEE Trans. Med. Imaging*, vol. MI-6, pp. 98–105, June 1987.

[4] J.H.A. Klotz and D.L. Snyder, "A hardware architecture using finite-field arithmetic for computing maximum-likelihood estimates in emission tomography," *IEEE Trans. Med. Imaging*, vol. 7, pp. 279–290, Dec. 1988.

[5] L. Kaufman, "Implementing and accelerating the EM algorithm for positron emission tomography," *IEEE Trans. Med. Imaging*, vol. MI-6, pp. 37–51, Mar. 1987.

[6] M.V. Ranganath, A.P. Dhawan, and N. Mullani, "A multigrid expectation maximization reconstruction algorithm for positron emission tomography," *IEEE Trans. Med. Imaging*, vol. 7, pp. 273–278, Dec. 1988.

[7] K. Lange and R. Carson, "Em reconstruction algorithms for emission and transmission tomography," *J. Comput. Assist. Tomogr.*, vol. 8, no. 2, pp. 306–316, Apr. 1984.

[8] E. Veklerov and J. Llacer, "Stopping rule for the MLE algorithm based on statistical hypothesis testing," *IEEE Trans. Med. Imaging*, vol. MI-6(4), pp. 313–319, Dec. 1987.

[9] J. Llacer and E. Veklerov, "Feasible image and practical stopping rules for iterative algorithms in emission tomography," *IEEE Trans. Med. Imaging*, vol. 8, pp. 186–193, June 1988.

[10] H.J. Trussell, "Convergence criteria for iterative restoration methods," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-31, pp. 129–136, Feb. 1983.

[11] C.K. Rushforth, "Signal restoration, functional analysis, and Fredholm integral equations of the first kind," in *Image Recovery: Theory and Applications*, H. Stark, Ed. Orlando, FL: Academic, 1987, ch. 1, pp. 1–27.

[12] G. Demoment, "Image reconstruction and restoration: overview of common estimation structures and problems," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 2024–2036, Dec. 1989.

[13] S. Kawata and O. Nalcioglu, "Constrained iterative reconstruction by the conjugate gradient method," *IEEE Trans. Med. Imaging*, vol. MI-4, pp. 65–71, June 1985.

[14] K. Tanabe, "Projection method for solving a singular system of linear equations and its applications," *Numerische Mathematik*, vol. 17, pp. 203–214, 1971.

[15] Y. Censor, "Finite series-expansion reconstruction methods," *IEEE Proc.*, vol. 71, pp. 409–419, Mar. 1983.

[16] L.A. Shepp and Y. Vardi, "Maximum likelihood reconstruction for emission tomography," *IEEE Trans. Med. Imaging*, vol. MI-1, pp. 113–122, Oct. 1982.

[17] D.G. Luenberger, *Optimization by Vector Space Methods*. New York: Wiley, 1968.

[18] L. Landweber, "An iterative formula for Fredholm integral equations of the first kind," *Amer. J. Math.*, vol. 73, pp. 615–624, 1951.

[19] O.N. Strand, "Theory and methods related to the singular-function expansion and Landweber's iteration for integral equations of the first kind," *Siam J. Numer. Anal.*, vol. 11, no. 4, pp. 798–825, 1974.

[20] T.-S. Pan and A.E. Yagle, "Numerical study of multigrid implementations of some iterative image reconstruction algorithms," *IEEE Trans. Med. Imaging*, vol. 10, pp. 572–588, Dec. 1991.

[21] I.M. Johnstone and B.W. Silverman, "Speed of estimation in positron emission tomography and related inverse problems," *Ann. Statist.*, vol. 18, no. 1, pp. 251–280, 1990.

[22] M.F. Smith, C.E. Floyd, and R.E. Coleman, "Reconstruction of SPECT images using generalized matrix inverses," in *Symp. IEEE Nucl. Sci.*, Arlington, VA, 1990.

[23] B. Nobel and J.W. Daniel, *Applied Linear Algebra*. Englewood Cliffs, NJ: Prentice-Hall, 1988, 3rd ed.

[24] H.J. Trussell and M.R. Civanlar, "The Landweber iteration and projection onto convex sets," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, pp. 1632–1634, Dec. 1985.

[25] J.L.C. Sanz and T.S. Huang, "Unified Hilbert space approach to iterative least-squares linear signal restoration," *J. Opt. Soc. Amer.*, vol. 73, no. 11, pp. 1455–1465, Nov. 1983.

[26] H.H. Barrett, J.N. Aarsvold, and T.J. Roney, "Null functions and eigenfunctions: tools for analysis of imaging systems," in *Information Processing in Medical Imaging*, 1991, pp. 211–226.

[27] U.G. Rothblum and C.P. Tan, "Upper bounds on the maximum modulus of subdominant eigenvalues of nonnegative matrices," *Linear Algebra Appl.*, vol. 66, pp. 45–86, 1985.

[28] A. Jennings, *Matrix Computation for Engineers and Scientists*. New York: Wiley, 1977.

[29] T.-S. Pan, "The generalized Landweber iteration in positron emission tomography," Ph.D. dissertation, Dep. Elec. Eng. Comput. Sci., The University of Michigan, Ann Arbor, MI, 1991.

[30] A.C. Kak and M. Slaney, *Principles of Computerized Tomographic Imaging*. New York: IEEE Press, 1988.

[31] J.A. Stamos, W.L. Rogers, N.H. Clinthorne, and K.F. Koral, "Object-dependent performance comparison of two iterative reconstruction algorithms," *IEEE Trans. Nucl. Sci.*, vol. 35, pp. 611–614, Feb. 1988.

[32] T.F. Budinger, G.T. Gullberg, and R.H. Huesman, "Emission computed tomography," in *Reconstruction from Projections: Implementation and Applications*, G.T. Herman, Ed. New York: Springer-Verlag, 1979, ch. 5, pp. 147–246.

APPENDIX G2

T.-S. Pan and A.E. Yagle, "Acceleration of Landweber-Type Algorithms by Suppression of Projection on the Maximum Singular Vector," IEEE 1991 Medical Imaging Conference, Santa Fe, NM, Nov. 5-9, 1991, pp. 2023-2027.

This is the conference paper version of Appendix G1.

Acceleration of Landweber-Type Algorithms by Suppression of Projection on the Maximum Singular Vector

Tin-Su Pan *and Andrew E. Yagle
Department of Electrical Engineering and Computer Science
The University of Michigan, Ann Arbor, MI 48109-2122

ABSTRACT

We develop a new procedure that speeds up the convergence during the initial stage (the first 100 forward and backward projections) of Landweber-type algorithms for iterative image reconstruction, which include the Landweber, generalized Landweber, and steepest descent algorithms. The procedure first identifies the singular vector associated with the maximum singular value of the PET system matrix, and then suppresses projection of the data on this singular vector after a single Landweber iteration. We show that typical PET system matrices have a significant gap between their two largest singular values; hence this suppression allows larger gains in subsequent iterations, speeding up convergence by roughly a factor of three. New contributions of this paper include: 1) study of the singular value spectra of typical PET system matrices; 2) study of the effect on convergence of projection on the maximum singular vector; and 3) study of the convergence behavior of the new procedure applied to the Landweber, generalized Landweber, steepest descent, conjugate gradient, and ART algorithms (comparison is also made with the MLEM algorithm).

I. INTRODUCTION

Some of the most researched algorithms in iterative image reconstruction in emission tomography are maximum-likelihood EM (MLEM) [1], algebraic reconstruction technique (ART) [2], steepest descent (STP) [3], and conjugate gradient (CG) [3]. One characteristic with the above algorithms is that they are all based on optimization strategies, which lead the reconstruction to be dependent on the object, such that no formula is able to describe what has been achieved after some number of iterations.

ART modifies the reconstructed image by projecting from one hyperplane to another hyperplane defined by a system of linear equations; MLEM maximizes a likelihood function; and STP and CG search adaptively for the largest gradient defined by the least-squares error between the projection data and the estimated projection data. In each case, the precise meaning of the image following each iteration is unclear.

One alternative is to use the generalized Landweber iteration [4], a method which can guarantee a characterized property of the reconstructed image after a number of iterations. Using this iteration [5], the reconstruction process is treated as a *filtering in the singular space* (defined by the singular values and singular vectors of the system matrix) [4]; no matter what the object is, the reconstructed image possesses a predefined characteristic after each single iteration, which can be designed before the iteration starts. Moreover, with a proper shaping matrix, the iteration can accelerate and regularize the image reconstruction process. Since the iteration is system dependent, not object dependent, computation time can be estimated accurately for various objects.

We applied the procedure to the image reconstruction of a positron emission tomography (PET) system. The procedure first identifies the singular vector associated with the maximum singular value of the PET system matrix, and then *suppresses* the projection of the image on this singular vector after a single Landweber iteration. The projection is defined as the DC-component of the image. We show that typical PET system matrices have a significant gap between their two largest singular values (this gap is inherent in most matrices with non-negative coefficients [6]); hence this suppression allows larger gain factors in the subsequent iterations, speeding up convergence by roughly a factor of three.

This paper is organized as follows. Section II formulates the image reconstruction problem as the solution to a large system of linear equations, and reviews the Landweber-type iterations. Section III introduces the new DC-suppression procedure. Section IV presents and summarizes numerical results. Comparisons with the MLEM, ART and CG iterations (both with and without applying the DC-suppression procedure) are also included. Section V concludes with a summary.

*The authors wish to thank Dr. W. Leslie Rogers and Mr. Neal H. Clinthorne of the Division of Nuclear Medicine of the University of Michigan for many helpful discussions. The work of the first author was supported in part by NIH grant #PO1-CA42768, and in part by a Research Partnership award from the H. H. Rackham School of Graduate Studies of the University of Michigan. The work of the second author was supported in part by ONR grant #N00014-90-J-1897.

0-7803-0513-2/92$0.3.00 ©IEEE

II. BACKGROUND

A. System Equation

The system equation that describes the transformation or projection process in a PET system is usually represented as a linear system $Ax = b$, where A is an $m \times n$ system matrix, which describes the system geometry, x is an $n \times 1$ vector of the image pixels, and b is an $m \times 1$ vector of the measured projection of the image. There are several problems associated with solving the system: 1) typically m and n are on the order of several thousand, which makes a direct (as opposed to iterative) approach to solve it very difficult; 2) A has a large condition number, which causes a significant change in the solution $x^* = A^\dagger b$, where $A^\dagger$ is the pseudoinverse of A, from a small perturbation in b; and 3) A is very sparse (about 2–3% non-sparse) [7].

B. Landweber-Type Algorithms

We define a *Landweber-type iteration* as an iteration of the form

$$x^{k+1} = x^k + \alpha_k D A^T (b - A x^k), \tag{1}$$

where α_k is a scalar and x^k is the reconstructed image after the k-th iteration. The iteration becomes:

1) the *Landweber iteration* [8] when $\alpha_k = \alpha$ (a constant) and $D = I$ (the identity matrix);

2) the *generalized Landweber iteration* [4] when $\alpha_k = \alpha$ (a constant) and D is the *shaping matrix*, a polynomial function of $\alpha A^T A$. D is assumed to be $F(\lambda = \alpha A^T A)$ [4] in this research:

$$\begin{aligned} F(\lambda) &= 31.5 - 315\lambda + 1443.75\lambda^2 - 3465\lambda^3 \\ &\quad +4504.5\lambda^4 - 3003\lambda^5 + 804.375\lambda^6; \end{aligned} \tag{2}$$

3) the *STP iteration* [3] when $D = I$ and α_k changes according to the largest gradient defined by the least-squares error between the projection data and the estimated projection data:

$$x^{k+1} = x^k + \alpha_k q_k, \tag{3}$$

$$\alpha_k = \frac{q_k^T q_k}{(Aq_k)^T A q_k}, \quad q_k = A^T(b - Ax^k). \tag{4}$$

For simplicity, we refer to both α and α_k as gain factors. When initialized with zero, all three iterations converge to x^* provided the Euclidean norm $\|\alpha D A^T A\|_2 < 2$ [4] for the (generalized) Landweber iteration, whose α is normally set to $\frac{1}{\sigma_1^2}$.

III. THE DC-SUPPRESSION PROCEDURE

It was observed [5] that 1) the DC-component of the image could be recovered after one single Landweber iteration when $\alpha = \frac{1}{\sigma_1^2}$; 2) the difference in magnitude between the two largest singular values σ_1 and σ_2 of a PET system matrix is large. Fig. 1 shows one example of the distribution of the singular values of the synthetic PET system matrix A_1 in [9], where σ_1 is scaled to 1 and the other σ's are scaled accordingly. In this example, σ_1 is much larger than σ_2 ($\sigma_1 = 1.0433$, $\sigma_2 = 0.61995$, the ratio $r_\sigma = \frac{\sigma_1}{\sigma_2} = 1.683$); and 3) the speed of the Landweber-type iteration is limited if the DC-component stays in the iteration. Therefore, it is reasonable that after applying one single Landweber iteration and removing the DC-component from the iteration, the Landweber-type iteration can be accelerated by increasing α to $\frac{1}{\sigma_2^2}$ (a larger gain!). The DC-component will be added back after the iteration is over. It was shown [5] that σ_1 and σ_2 and their associated singular vectors v_1 and v_2 can be derived from using the power method and a purification procedure [10] without much effort, which can be carried out during data acquisition process. Fig. 2 shows the flowchart of the DC-suppression procedure. Note that x_{DC} denotes the DC-component.

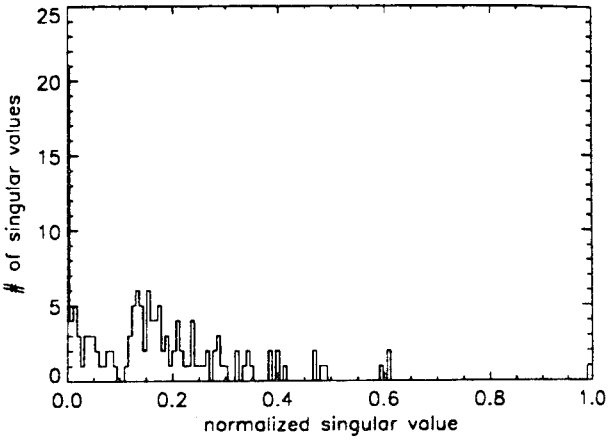


Figure 1: An example of the singular value histogram of a PET system. Note the singular values have been normalized or divided by their largest singular value. There are 21 singular values very close to zero.

To show how the DC-suppression procedure accelerates convergence, let $r_\sigma = \frac{\sigma_1}{\sigma_2} = 1.68$, which approximates the value r_σ of the system matrix A_3 [9] whose dimension is 4160×3228, $\sigma_1 = 0.986703$, and $\sigma_2 = 0.587256$. Define multiplication of A by an image vector (of dimension n) as a *forward* projection, and multiplication of A^T by a projection data vector (of dimension m) as a *backward* projection. The derivation of σ_1 and σ_2 took 9 and 25 forward and backward projections, respectively. Most computation in iterative reconstruction algorithms consists of these two operations.

The computation for DC suppression consists of one forward and backward projection (see the first four blocks in Fig. 2). A single Landweber iteration also requires one forward and backward projection. The generalized Landweber iteration, with $D = F(\alpha A^T A)$ as in (2), requires 7

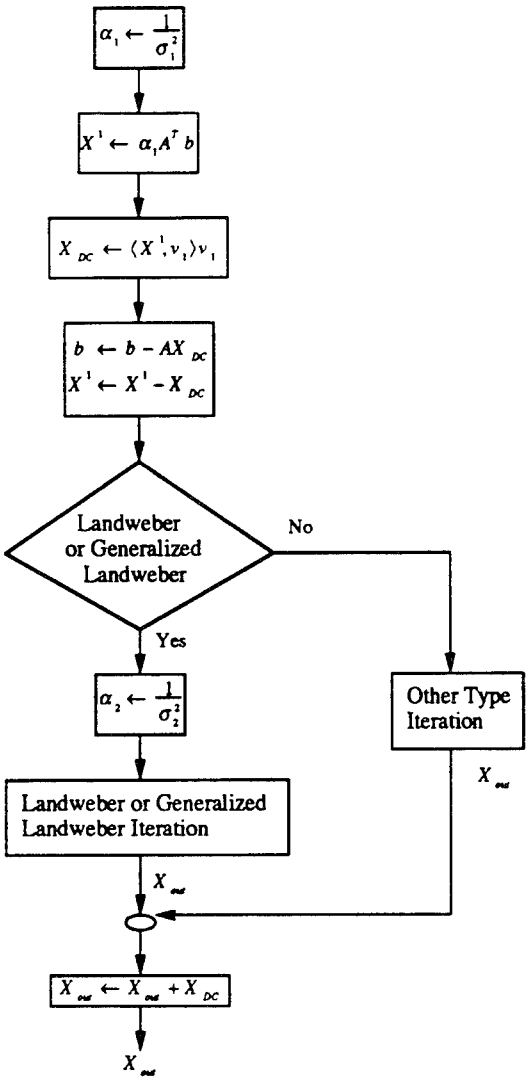


Figure 2: The flowchart of the DC-suppression procedure.

forward and backward projections per iteration [5].

Fig. 3 shows the number of forward and backward projections needed to recover 95% of image components on various singular vectors when $r_\sigma = 1.68$ for the Landweber and generalized Landweber iterations both with and without DC-suppression. Without loss of generality, the largest singular value is assumed to be 1. Fig. 3 shows that after about 22 forward and backward projections (1 from the DC suppression and 21 from 3 generalized Landweber iterations): 1) the generalized Landweber iteration recovers more components than the Landweber iteration; 2) the (generalized) Landweber iteration with DC suppression recovers more high-frequency components than the (generalized) Landweber iteration alone; and 3) the generalized Landweber iteration with DC suppression recovers all image components on singular vectors with singular values greater than 0.1; to achieve this without DC suppression, 63 forward and backward projections are necessary. Hence the Landweber and generalized Landweber iterations without DC suppression require roughly *three times* as many

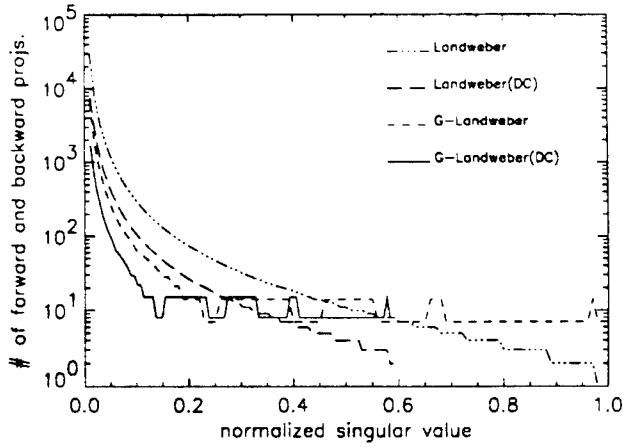


Figure 3: The number of forward and backward projections needed for the recovery of 95% of components on various singular vectors. Without loss of generality, the largest singular value is assumed to be 1. The symbol DC after Landweber means Landweber with DC suppression; the G-Landweber means generalized Landweber. Note that $r_\sigma = \frac{\sigma_1}{\sigma_2} = 1.68$.

projections to reach the same convergence point as the iterations with DC suppression. This factor-of-three speed-up, together with the possibility of parallel implementation, suggests greater application of these algorithms in the future.

IV. Simulation Results

The Landweber-type algorithms, ART and CG, all with and without the DC-suppression procedure, were compared in the reconstruction of the image of the *complex* phantom (see Fig. 4) in the PET system with system matrix A_3 [5]. The ART and MLEM iterations both with average initial condition [1] (each initial image pixel is set to be the total value of b divided by the total number of image pixels) were also included for comparison.

Fig. 5 shows the results. Each symbol in a curve represents the result after a single iteration. The comparison was based on how close the solution to the true image *vs.* the number of forward and backward projections spent. It is clear that the Landweber-type algorithms were all accelerated using the DC-suppression procedure (see Fig. 5(a) and (c)). Fig. 6 explains that the acceleration of STP with the DC-suppression procedure was directly related to the larger gains caused by the DC suppression during the first 12 forward and backward projections. ART with the DC-suppression procedure was faster than ART without the DC-suppression but was comparable to ART with average initial condition (see Fig. 5(b)). This indicates that ART can be significantly accelerated by simply using the aver-

[1] a common initial condition setting in emission tomography.

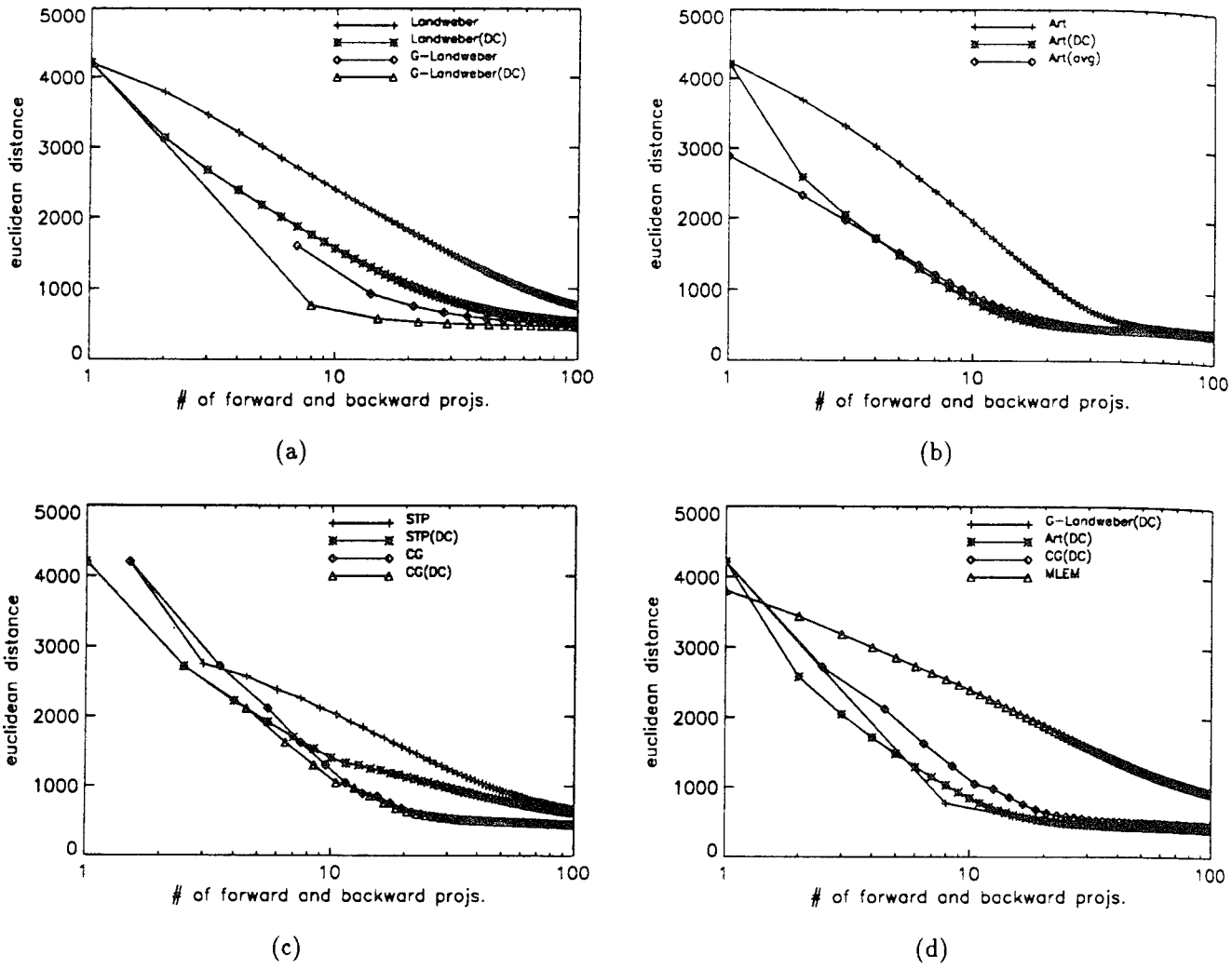


Figure 5: The performance of reconstructing the image of the complex phantom using (a) the Landweber iteration, the generalized Landweber iteration, and their DC suppression implementations; (b) ART iterations with zero, average, and DC suppression implementation; (c) STP iteration, CG iteration, and their DC suppression implementations; and (d) the fastest methods in the previous three figures (one from each figure) and MLEM iteration.

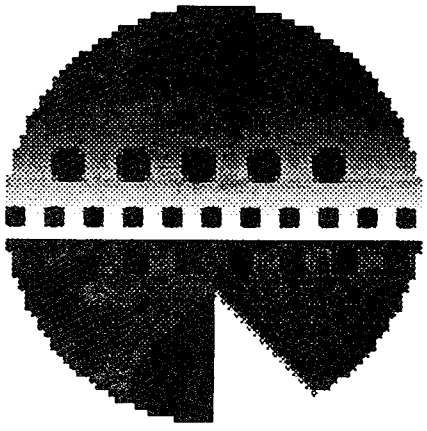


Figure 4: The image of the complex phantom.

age initial condition for the reconstruction of the complex phantom whose values are non-negative. In this example, ART can be accelerated about three times by simply using average initial condition. CG could not be accelerated (see Fig. 5(c)). The generalized Landweber iteration was faster than CG and was comparable to ART, all with the DC-suppression procedure (see Fig. 5(d)).

V. Summary

A new DC-suppression procedure has been developed, and shown to significantly accelerate the convergence of the Landweber-type iterations. The procedure identifies and removes the DC component from the iteration, permitting larger gains α_k and accelerating convergence. The convergence speed-up is due to the large gap between the two largest singular values of the system matrix; this gap was

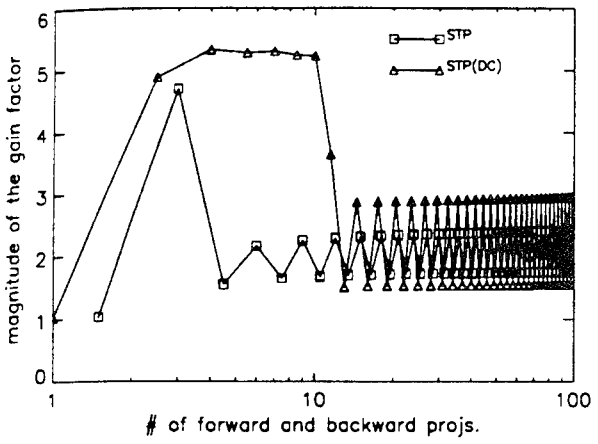


Figure 6: The change of the gain factors in STP iteration after DC suppression.

observed in PET system matrices of different scales, and is likely to be present in any problem with a non-negative system matrix.

The generalized Landweber iteration with DC suppression recovered 95% of the image components on singular vectors with singular values greater than 0.1 in about 22 forward and backward projections when the ratio $r_\sigma = 1.68$; without DC suppression 63 forward and backward projections were necessary. Although ART with DC suppression showed faster convergence than ART with a zero initial condition, this was not a significant improvement over ART with an average initial condition. With DC suppression, generalized Landweber and ART converge faster than CG and MLEM (DC suppression can not be used on MLEM) during the initial stage of the iteration (the first 100 forward and backward projections). The generalized Landweber iteration with DC suppression was comparable to ART with DC suppression. However, considering the feasibility of parallel computing of the forward and the backward projections, and the controllability of the generalized Landweber iteration, the generalized Landweber iteration may be more favorable compared to ART.

Topics for further research include further study of the ratio r_σ for different system matrices, and investigation of the possibility of using different shaping matrices to speed up the generalized Landweber iteration with DC suppression.

REFERENCES

[1] L. A. Shepp and Y. Vardi, "Maximum likelihood reconstruction for emission tomography," *IEEE Trans. Medical Imaging*, vol. MI-1(2), pp. 113–122, Oct. 1982.

[2] Y. Censor, "Finite series-expansion reconstruction methods," *IEEE Proceedings*, vol. 71(3), pp. 409–419, Mar. 1983.

[3] S. Kawata and O. Nalcioglu, "Constrained iterative reconstruction by the conjugate gradient method," *IEEE Trans. Medical Imaging*, vol. MI-4(2), pp. 65–71, June 1985.

[4] O. N. Strand, "Theory and methods related to the singular-function expansion and Landweber's iteration for integral equations of the first kind," *Siam J. Numer. Anal.*, vol. 11(4), pp. 798–825, Sept. 1974.

[5] T.-S. Pan, *The Generalized Landweber Iteration in Positron Emission Tomography*. The University of Michigan: Dissertation, 1991.

[6] U. G. Rothblum and C. P. Tan, "Upperbounds on the maximum modulus of subdominant eigenvalues of nonnegative matrices," *Linear Algebra and its Applications*, vol. 66, pp. 45–86, 1985.

[7] L. Kaufman, "Implementing and accelerating the EM algorithm for positron emission tomography," *IEEE Trans. Medical Imaging*, vol. MI-6(1), pp. 37–51, Mar. 1987.

[8] L. Landweber, "An iterative formula for Fredholm integral equations of the first kind," *Am. J. Math.*, vol. 73, pp. 615–624, 1951.

[9] T.-S. Pan and A. E. Yagle, "Numerical study of multigrid implementations of some iterative image reconstruction algorithms," *to appear on IEEE Trans. Med. Imaging*, Dec. 1991.

[10] A. Jennings, *Matrix Computation for Engineers and Scientists*. New York: Wiley, 1977.

APPENDIX H1

T.-S. Pan and A.E. Yagle, "Numerical Study of Multigrid Implementations of Some Iterative Image Reconstruction Algorithms," IEEE Trans. Medical Imaging 10(4), 572-588, December 1991.

This paper investigates the use of several iterative algorithms, including the generalized Landweber iteration, in multigrid image reconstruction. The image is first reconstructed quickly on a coarse grid. This coarse image is then used as the initialization for reconstruction of the image on a fine grid. Many numerical examples are used to illustrate the performance of various algorithms.

Numerical Study of Multigrid Implementations of Some Iterative Image Reconstruction Algorithms

Tin-Su Pan, *Student Member, IEEE*, and Andrew E. Yagle, *Member, IEEE*

Abstract—**The numerical behavior of multigrid implementations of the Landweber, generalized Landweber, ART, and MLEM iterative image reconstruction algorithms is investigated. Comparisons between these algorithms, and with their single-grid implementations, are made on two small-scale synthetic PET systems, for phantom objects exhibiting different characteristics, and on one full-scale synthetic system, for a Shepp–Logan phantom. We also show analytically the effects of noise and initial condition on the generalized Landweber iteration, and note how to choose the shaping operator to filter out noise in the data, or to enhance features of interest in the reconstructed image. Original contributions include 1) numerical studies of the convergence rates of single-grid and multigrid implementations of the Landweber, generalized Landweber, ART, and MLEM iterations and 2) effects of noise and initial condition on the generalized Landweber iteration, with procedures for filtering out noise or enhancing image features.**

I. Introduction

ITERATIVE image reconstruction algorithms have been studied intensively in the last decade, for application in single photon emission computerized tomography (SPECT) and positron emission tomography (PET) [1]–[9]. Some believe [10], [11] that an iterative reconstruction algorithm produces a less noisy image than filtered back-projection (FBP) [12]. Some advantages of using an iterative algorithm are 1) iterative algorithms can still be applied when complete projection data are unavailable, 2) there is considerable control over the reconstruction process, and 3) spatially varying attenuation corrections can be incorporated [10], [11]. Some possible drawbacks of iterative algorithms are: 1) a large amount of computation and time required; and 2) the need to incorporate regularization [13]–[15] to make the iteration numerically stable and insensitive to noise.

Most iterative algorithms exhibit a nonuniform convergence property [3], [15]: *low-frequency components of the image tend to be recovered earlier in the iteration than high-frequency components*. Here high-frequency components are those associated with small singular-values, and low-frequency components are associated with large singular-values of the system matrix. This terminology is common (e.g., [15]), although these frequencies coincide with Fourier frequencies only in a special circumstance [19]. The convergence rate of the algorithm is thus limited by the relatively slow convergence of the high-frequency components of the image. Algorithms such as the Landweber iteration [16], algebraic reconstruction technique (ART) [2], and maximum-likelihood-expectation-maximization (MLEM) [5], allow little control over their convergence behavior.

The *generalized* Landweber iteration of Strand [17], in contrast, permits considerable control over its convergence rate by the choice of the shaping operator. Although it has not been generally used in PET imaging, this algorithm has several advantages: 1) it is parallelizable (unlike ART), permitting fast implementations, e.g., a forward or backward projection can be computed by multiple processors at the same time, 2) its convergence behavior can be specified using the shaping operator, and 3) the shaping operator can be used to filter out noise in the data or to enhance features in the image. These advantages will be discussed in Section III, along with details on the effects of noise and initial condition on the convergence rate of the iteration. In particular, we treat the data and noise separately, and reveal an important distinction between the effects of data and noise on the reconstructed image as the iteration progresses.

Initialization of an iterative algorithm has an important effect on its convergence. Initialization can be as simple as setting all pixels to zero, or as complex as using the result of FBP [1], [14]. The presumption behind the latter approach is that FBP should furnish a starting point that is "close" to the desirable image, after which the iteration would quickly converge. Surprisingly, this approach has shown little success [14]. In Section V, our numerical results lead to a possible explanation of why this is so. We also examine the effects of different initial conditions on the convergence behavior of different iterative algorithms.

The most important new contribution of this paper is a numerical study of *multigrid* implementations of various iterative algorithms: Landweber, generalized Landweber, ART, and MLEM. In a multigrid implementation, the iteration is first used on a coarse grid until it converges; the result is then interpolated and used as an initial condition on a fine grid. The coarse grid iteration requires much less computation per iteration. Although two different system geometry matrices are needed (one for each grid), both

Manuscript received July 23, 1990; revised June 6, 1991. The work of T.-S. Pan was supported in part by NIH under Grant PO1-CA42768. The work of A. E. Yagle was supported by the Office of Naval Research under Grant N00014-90-J-1897.

The authors are with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109-2122.

IEEE Log Number 9102716.

0278-0062/91$01.00 © 1991 IEEE

matrices are relatively sparse, so the additional storage required for the (much smaller) coarse-grid system matrix is minor. More details are given in Section IV.

We define a local smoothness property: *the values of four neighboring fine-grid pixels to be grouped as a coarse-grid pixel are close to each other.* Our results indicate that if the image has this property, then the convergence of high-frequency components of the image can be significantly accelerated using a multigrid implementation. "Local smoothness" is not strictly defined here, nor does it need to—the more this property holds, the greater the acceleration of convergence. If the local smoothness property does not hold, or if there are no high-frequency components in the image, then a multigrid implementation does not seem to speed up the convergence rate. Ranganath *et al.* [22] proposed a multigrid implementation of the MLEM algorithm, and gave a numerical example. In the present paper, results for several different algorithms are given, more examples are given, and more conclusions about reconstruction behavior are made.

Our numerical experiments were mainly conducted on two small-scale synthetic PET systems (see Section V for details). The systems are large enough to exhibit behavior that would be seen in real PET systems, but small enough to study in detail aspects of the image and the system (e.g., singular-values) that cause various types of convergence behavior. A numerical example of a large-scale synthetic PET system with a Shepp–Logan phantom [12] is also included. Many numerical experiments were performed; the examples given here are intended to be *illustrative*, not comprehensive, and our conclusions are *not* based just on the examples presented, but on many others as well.

The paper is organized as follows. Section II formulates the image reconstruction problem as the solution to a large system of linear equations, and gives a review of singular value decomposition (SVD). Section III studies analytically and in detail the convergence behavior of the Landweber and generalized Landweber iterations. The different roles of data, noise, and initial condition on the convergence rate are studied for the first time, and procedures for filtering out noise and enhancing image features are discussed. Section IV summarizes the idea of a multigrid implementation and presents the multigrid implementation adopted in this work. Section V presents, summarizes, and discusses numerical results, and presents some conclusions about convergence rates of multigrid implementations. Section VI concludes with a summary.

II. Problem Formulation

A. System Equation

The system equation [2], [5] that describes the transformation or projection process in SPECT or PET is usually represented as

$$Ax = b \tag{1}$$

where A is an $m \times n$ system matrix, which describes the system geometry, x is an $n \times 1$ vector of the image pixels, and b is an $m \times 1$ vector of the measured projection of the image. The matrix A is typically sparse [6].

There are several problems associated with the solution of (1).

1) The size of the system—typically m and n are in the order of thousands [5], [6].

2) Ill-conditioning of the system—typically A has a large condition number [15], so that a small change in the projection data b may cause a large change in the solution x or in the minimum-norm least-squares solution x^*,

$$x^* = A^\dagger b \tag{2}$$

where $A^\dagger$ is the pseudoinverse of A [24].

3) Ill-posedness—typically, A may have a nonempty null space, so that some components of the image can not be recovered from the projection b without additional information.

B. Review of Singular Value Decomposition

Let the singular value decomposition (SVD) [24] of the real $m \times n$ system matrix A be

$$A = U\Sigma V^T \tag{3}$$

where $U = [u_1, u_2, \cdots, u_m]$ and $V = [v_1, v_2, \cdots, v_n]$ are orthogonal matrices. The vectors u_i and v_i are singular-vectors, and U and V represent a set of orthonormal bases for the real Hilbert spaces $\mathcal{R}^m$ and $\mathcal{R}^n$, respectively. The matrix Σ is $m \times n$ and "diagonal" in that $\langle \Sigma \rangle_{i,j} = 0$ unless $i = j$, in which case $\langle \Sigma \rangle_{i,j} = \sigma_i$, the singular-values of A. The number of nonzero singular-values is the rank $\rho(A)$ of A [13], [17]. Without loss of generality, assume the maximum singular-value of A is one (this can always be done by scaling A) and let $1 = \sigma_1 \geq \sigma_2 \cdots > \sigma_{\rho(A)} > 0 = \sigma_{\rho(A)+1} = \cdots = \sigma_{\min\{m,n\}}$ where $\min\{m, n\}$ is the minimum of m and n. Also define $\sigma_{\min(m,n)+1} = \cdots = \sigma_{\max\{m,n\}} = 0$ where $\max\{m, n\}$ is the maximum of m and n.

It can be shown [13], [17] that

$$(AA^T)u_i = \sigma_i^2 u_i, \qquad i = 1, \cdots, m, \tag{4}$$

$$(A^TA)v_i = \sigma_i^2 v_i, \qquad i = 1, \cdots, n, \tag{5}$$

$$A^T u_i = \sigma_i v_i, \qquad i = 1, \cdots, m, \tag{6}$$

$$Av_i = \sigma_i u_i, \qquad i = 1, \cdots, n, \tag{7}$$

so that the squares σ_i^2 of the singular-values are the eigenvalues of the matrices AA^T and A^TA. It can also be shown [24] that the minimum-norm least-squares solution to (1) is

$$x^* = V \operatorname{diag}\left[\frac{1}{\sigma_1} \cdots \frac{1}{\sigma_{\rho(A)}} 0 \cdots 0\right] U^T b$$

$$= \sum_{i=1}^{\rho(A)} \frac{1}{\sigma_i} (b, u_i) v_i \tag{8}$$

where $(b, u_i) = b^T u_i$, the inner product of vectors b and u_i.

Let $R(A)$ and $N(A)$ be the range and the null space of A, respectively, and let $R(A^T)$ and $N(A^T)$ be the range and the null space of A^T, respectively. It is also true [25] that

$$R(A) = \text{span}\ \{u_1, \cdots, u_{\rho(A)}\}, \tag{9}$$

$$N(A^T) = \text{span}\ \{u_{\rho(A)+1}, \cdots, u_m\}, \tag{10}$$

$$R(A^T) = \text{span}\ \{v_1, \cdots, v_{\rho(A)}\}, \tag{11}$$

$$N(A) = \text{span}\ \{v_{\rho(A)+1}, \cdots, v_n\}, \tag{12}$$

and

$$\mathcal{R}^m = R(A) \oplus N(A^T), \qquad \mathcal{R}^n = R(A^T) \oplus N(A) \tag{13}$$

where $\oplus$ means direct sum. In fact, $N(A^T) = R(A)^\perp$ and $R(A^T) = N(A)^\perp$ where $R(A)^\perp$ and $N(A)^\perp$ are the orthogonal complements of $R(A)$ and $N(A)$, respectively.

III. Nonuniform Convergence of the Landweber and Generalized Landweber Iterations

In this section, we review the Landweber and generalized Landweber iterations, and the nonuniform convergence behavior of these algorithms: low-frequency components of the image are recovered earlier in the iteration than high frequency components. We also make some new points about the effects of noise and initial condition on the convergence behavior of these algorithms. These points will be important in interpreting the numerical results in Section V.

A. Review of the Landweber and Generalized Landweber Iterations

The Landweber iteration method [16] was proposed in 1951. This iteration is

$$x^k = x^{k-1} + A^T(b - Ax^{k-1}) \tag{14}$$

where x^k, in our context, denotes the reconstructed image after the kth iteration, and the superscript T means transpose. This iteration will converge to the minimum-norm least-squares solution x^* if $\|A^TA\|_2 < 2$ and the iteration is initialized with $x^0 = 0$ [17], [18]. A formula showing the performance after each iteration will be derived for this iteration in Section III-B.

The Landweber iteration method was extended by Strand [17] to the generalized Landweber iteration method, which uses

$$x^k = x^{k-1} + DA^T(b - Ax^{k-1}) \tag{15}$$

for the iteration. The matrix D, called the shaping matrix, is a linear operator and can be designed as a polynomial function of A^TA [17] to emphasize some frequency components of an image and to accelerate the convergence of these components. In this paper, we also assume the matrix D is a polynomial function of A^TA. If $D = I$, then the generalized Landweber iteration in (15) reduces to the Landweber iteration in (14). The convergence of the iteration in (15) is assumed when $\|DA^TA\|_2 < 2$ [17]. It will be shown in Section III-C that use of a proper D matrix [17] results in the generalized Landweber iteration converging faster than the Landweber iteration. Note that the generalized Landweber iteration is similar to the Jansson Van Cittert (JVC) iteration [20], [21]. Differences between the algorithms are that the generalized Landweber iteration involves matrices, and D is restricted to be a linear operator. We do not consider nonlinear D operators in this paper.

The major advantage in using the generalized Landweber iteration in image reconstruction is that matrix D behaves as a filtering operator, so that the generalized Landweber iteration can selectively reconstruct or emphasize some frequency band of interest in an image, while attenuating other frequencies in the same image. The major disadvantage is the extra multiplication introduced by D, which depends proportionally on the degree of the polynomial function being used. However, the generalized Landweber iteration is parallelizable, unlike ART; this helps compensate for the extra computation.

B. Analysis of Convergence of the Landweber Iteration

From (1), any component of the image x in the null space $N(A)$ has no contribution to the projection b. In the (generalized) Landweber iteration with zero initial condition [see (14) and (15), and see (19) and (27) later], no component of the image x in $N(A)$ can be recovered through the iteration, which is dependent on b, and any component of the projection b not in $R(A)$ has no effect in the iteration.

Now suppose the projection data are noisy, so that

$$b = b_0 + \epsilon \tag{16}$$

where $b_0 \in R(A)$ is the actual value of the projection data, and the noise ϵ models measurement noise and Poisson counting noise. The noise ϵ and initial condition x_0 can be further decomposed as

$$\epsilon = \epsilon_0 + \epsilon_0^\perp, \tag{17}$$

$$x_0 = \tilde{x}_0 + x_0^\perp \tag{18}$$

where $\epsilon_0 \in R(A)$, $\epsilon_0^\perp \in R(A)^\perp$, $\tilde{x}_0 \in R(A^T)$ and $x_0^\perp \in R(A^T)^\perp = N(A)$. Notice that in (17) only ϵ_0 can affect the measurement of b_0, and only ϵ_0 may have an adverse effect in the iterations (14) and (15) since $A^T\epsilon_0^\perp = 0$.

We now make some new observations about the effects of noise and initial condition on the convergence behavior. By expressing the reconstructed image x^k in (14) as a function of the projection b and initial condition $x_0 = \tilde{x}_0 + x_0^\perp$ as in (18), and using (3), we can show

$$\begin{aligned} x^k = x_0^\perp &+ \sum_{i=1}^{\rho(A)} [1 - (1 - \sigma_i^2)^k] \frac{1}{\sigma_i} (b_0, u_i) v_i \\ &+ \sum_{i=1}^{\rho(A)} [1 - (1 - \sigma_i^2)^k] \frac{1}{\sigma_i} (\epsilon_0, u_i) v_i \\ &+ \sum_{i=1}^{\rho(A)} (1 - \sigma_i^2)^k (x_0, v_i) v_i. \end{aligned} \tag{19}$$

Equation (19) plays an important role in exploring how the singular-values σ_i and the number of iterations k affect the relationship between the reconstructed image x^k and the projection b_0, the noise ϵ_0, and the initial condition x_0.

In (19) let $k \to \infty$, and in (8) let $b = b_0 + \epsilon$ as in (16). Comparing these, the only difference is x_0^+, the component of the initial condition x_0 in the null space $N(A)$. Define

$$G_x(\sigma_i, k) \triangleq [1 - (1 - \sigma_i^2)^k], \tag{20}$$

$$G_{\epsilon_0}(\sigma_i, k) \triangleq [1 - (1 - \sigma_i^2)^k] \frac{1}{\sigma_i}, \tag{21}$$

$$G_{x_0}(\sigma, k) \triangleq (1 - \sigma_i^2)^k. \tag{22}$$

Then $G_x(\sigma_i, k)$, $G_{\epsilon_0}(\sigma_i, k)$, and $G_{x_0}(\sigma_i, k)$ represent the component gains of data, noise, and initial condition, respectively, after the first k iterations, on the singular-vector v_i associated with singular-value σ_i.

Note that the data gain $G_x(\sigma_i, k)$ does not include the factor $1/\sigma_i$, while the noise gain $G_{\epsilon_0}(\sigma_i, k)$ does. These gain definitions are made for the following reasons. The noise ϵ_0, causing the reconstructed image x^k in the Landweber iteration to differ from the noiseless minimum-norm least-squares solution of $Ax = b_0$, mainly comes from measurement or observation noise, not from Poisson counting noise. This noise does *not* arise in the projection process, which is low-pass in nature [26]; it is added *afterwards*, so that the factor $1/\sigma_i$ is lumped with (ϵ_0, u_i) in the noise gain $G_{\epsilon_0}(\sigma_i, k)$ (Poisson counting noise, which does come from the projection process, is assumed to have a limited effect on the reconstructed image x^k). On the other hand, since the projection data b_0 *does* come from the projection process, the data gain $G_x(\sigma_i, k)$ should not include $1/\sigma_i$ as we compare (8) and (19).

Equation (19) illustrates several points.

1) The projection b_0, noise ϵ_0, and initial condition x_0 can be decomposed into components each having an independent contribution to and a different convergence rate in the resulting reconstruction.

2) The data gain $G_x(\sigma, k)$ has the property

$$\lim_{k \to \infty} G_x(\sigma, k) = 1, \qquad 0 < \sigma \leq 1, \tag{23}$$

so that all components of the image not in $N(A)$ will eventually be recovered.

3) However, the noise gain $G_{\epsilon_0}(\sigma, k)$ will also become larger and larger as the iteration progresses (i.e., as the index k increases):

$$\lim_{k \to \infty} G_{\epsilon_0}(\sigma, k) = \frac{1}{\sigma}, \qquad 0 < \sigma \leq 1. \tag{24}$$

Comparing (23) and (24) and noting that A usually has some very small singular-values, the noise will be greatly amplified by the time the image components in $R(A^T)$ are completely recovered since $1/\sigma_i >> 1$ for small σ_i. This is due to the ill-conditioning of the system. A noise component on the singular-vector v_i will be amplified by a factor proportional to $1/\sigma_i$.

4) The initial condition gain $G_{x_0}(\sigma, k)$ converges to zero

$$\lim_{k \to \infty} G_{x_0}(\sigma, k) = 0, \qquad 0 < \sigma \leq 1 \tag{25}$$

so that the initial condition component $\bar{x}_0$ will become less and less important as the iteration progresses.

5) However, the vector x_0^+ produces a term independent of the iteration. Hence, a poor initial condition can impose a bias on the reconstruction which can not be corrected by the iteration.

Fig. 1 shows the convergence properties of the data gain $G_x(\sigma, k)$, noise gain $G_{\epsilon_0}(\sigma, k)$ and initial condition gain $G_{x_0}(\sigma, k)$ for the first 30 Landweber iterations. In Fig. 1, the data components on singular-vectors with larger singular-values are recovered faster, and components on singular-vectors with smaller singular-values are recovered relatively slower. This demonstrates the nonuniform convergence property: the reconstruction of high-frequency components converges slowly, while the reconstruction of low-frequency components converges relatively fast. Here the "high" and "low" frequencies of an image are defined as the components on singular-vectors with "small" singular-values and the components on singular-vectors with "large" singular-values, respectively.

C. Analysis of Convergence of the Generalized Landweber Iteration

From the generalized Landweber iteration in (15), it can be seen that D is an operator mapping $R(A^T)$ to $\Re^n$. Since $R(A^T)$ is spanned by $\{v_1, \cdots, v_{\rho(A)}\}$ as in (11), the definition of D will be complete if the image under Dv_i of each singular-vector v_i in $R(A^T)$ is specified [17]. Strand [17] proposed that a matrix D could be designed by specifying the scalars $p_1, \cdots, p_{\rho(A)}$ in

$$Dv_i = p_i v_i, \qquad 0 < p_i \sigma_i^2 < 2 \tag{26}$$

where the condition $0 < p_i \sigma_i^2 < 2$ assures the convergence of the generalized Landweber iteration. Repeating the procedure for deriving (19), the reconstructed image x^k after the kth generalized Landweber iteration will be

$$\begin{aligned} x^k = x_0^+ &+ \sum_{i=1}^{\rho(A)} [1 - (1 - p_i \sigma_i^2)^k] \frac{1}{\sigma_i} (b_0, u_i) v_i \\ &+ \sum_{i=1}^{\rho(A)} [1 - (1 - p_i \sigma_i^2)^k] \frac{1}{\sigma_i} (\epsilon_0, u_i) v_i \\ &+ \sum_{i=1}^{\rho(A)} (1 - p_i \sigma_i^2)^k (x_0, v_0) v_i. \end{aligned} \tag{27}$$

The first sum appeared in [17]; the additional sums representing the effects of noise and initial condition are new. Define [compare to (20)–(22)]

$$G_x'(\sigma_i, k) \triangleq [1 - (1 - p_i \sigma_i^2)^k], \tag{28}$$

$$G_{\epsilon_0}'(\sigma_i, k) \triangleq [1 - (1 - p_i \sigma_i^2)^k] \frac{1}{\sigma_i}, \tag{29}$$

$$G_{x_0}'(\sigma_i, k) \triangleq (1 - p_i \sigma_i^2)^k. \tag{30}$$

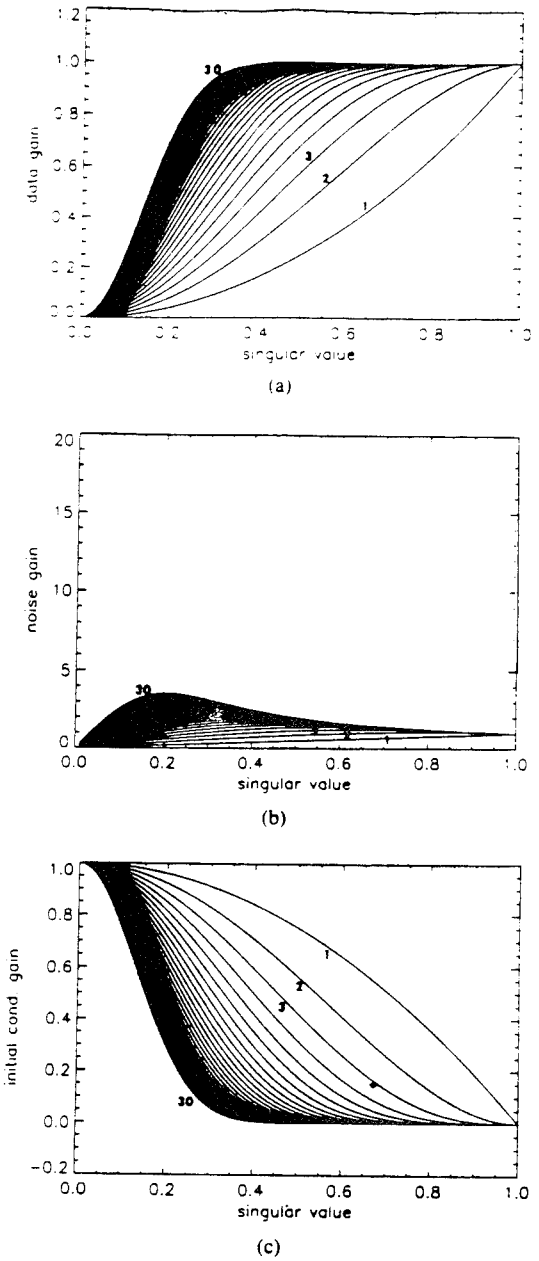


Fig. 1. The convergence properties of (a) data gain $G_x(\sigma, k)$, (b) noise gain $G_{\epsilon_0}(\sigma, k)$, and (c) initial condition gain $G_{x_0}(\sigma, k)$ versus singular-value σ for the first 30 Landweber iterations. The first, second, and third iteration plus the last iteration is labeled.

Then $G'_x(\sigma_i, k)$, $G'_{\epsilon_0}(\sigma_i, k)$, and $G'_{x_0}(\sigma_i, k)$ represent the component gains of data, noise, and initial condition, respectively, after the first k iterations on the singular-vector v_i with singular-value σ_i. The convergence properties of $G'_x(\sigma, k)$, $G'_{\epsilon_0}(\sigma, k)$, and $G'_{x_0}(\sigma, k)$ are [compare to (23)–(25)]

$$\lim_{k \to \infty} G'_x(\sigma, k) = 1, \qquad 0 < \sigma \leq 1, \tag{31}$$

$$\lim_{k \to \infty} G'_{\epsilon_0}(\sigma, k) = \frac{1}{\sigma}, \qquad 0 < \sigma \leq 1, \tag{32}$$

$$\lim_{k \to \infty} G'_{x_0}(\sigma, k) = 0, \qquad 0 < \sigma \leq 1, \tag{33}$$

respectively.

From (23)–(25) and (31)–(33), we see that the asymptotic behavior of the Landweber iteration is identical to that of the generalized Landweber iteration. The only difference between (19) and (27) is σ_i^2 in (19) has been changed to $p_i \sigma_i^2$ in (27). When $p_i = 1$, the iteration in (27) is identical to that in (19). We also see that choosing p_i can modify the convergence rates of the data gain $G'_x(\sigma_i, k)$, noise gain $G'_{\epsilon_0}(\sigma_i, k)$, and initial condition gain $G'_{x_0}(\sigma_i, k)$. However, the convergence rates of $G'_x(\sigma_i, k)$, $G'_{\epsilon_0}(\sigma_i, k)$ and $G'_{x_0}(\sigma_i, k)$ will all be accelerated or be decelerated at the same time.

We now demonstrate the convergence behavior of the generalized Landweber iteration for two different choices of the shaping matrix D. First, let D be specified by choosing $p_i = 1/\sigma_i^2$, $\sigma_i \neq 0$ in (27). Then the limiting behavior described in (31)–(33) is attained after a single iteration! Of course, (27) along with zero initial condition and $p_i = 1/\sigma_i^2$ make it clear that this choice of D is equivalent to the direct computation of the minimum-norm least-squares solution x^* in (8).

The second choice of D is from [17], in which D is given as a polynomial function $F(\cdot)$ of A^TA, i.e., $D = F(A^TA)$ where

$$F(\lambda) = 31.5 - 315\lambda + 1443.75\lambda^2 - 3465\lambda^3 + 4504.5\lambda^4 - 3003\lambda^5 + 804.375\lambda^6. \tag{34}$$

The polynomial function $F(\lambda)$ is chosen so that $\lambda F(\lambda)$ is a good approximation to the unit step function. This choice results [see (5) and (26)] $Dv_i = F(A^TA)v_i = F(\sigma_i^2)v_i$ and $F(\sigma_i^2) = p_i$, so that (34) effectively chooses $p_i\sigma_i^2 \approx 1$ to achieve the minimum-norm least-squares solution faster [see (27)]. Hereafter, we will assume this polynomial function is used for the D matrix in the generalized Landweber iteration. From this example, we can see that it is not necessary to carry out an SVD computation in order to design the shaping matrix D.

Fig. 2 shows the convergence properties of the data gain $G'_x(\sigma, k)$, noise gain $G'_{\epsilon_0}(\sigma, k)$ and initial condition gain $G'_{x_0}(\sigma, k)$ for the first 30 generalized Landweber iterations using (34). Comparing the convergence patterns in Figs. 1 and 2, we see that using (34) in the generalized Landweber iteration accelerates the convergence rate (in terms of the number of iterations), and further attenuates the effect of initial condition, but at the same time further amplifies the noise.

D. Filtering with the Generalized Landweber Iteration

Equations (27)–(30) show that the generalized Landweber iteration can *filter* either the image or the data, reducing the effects of noise or enhancing features in the image. To see this, suppose we know *a priori* that the noise ϵ_0 lies (or is likely to lie) in some subspace spanned by $\{u_i, i \in \mathcal{S}\}$ where $\mathcal{S} \subset \{1, \cdots, \rho(A)\}$, and the component of b_0 lying in this subspace is not significant. Then we would like to eliminate the terms in the second sum of (27) with $i \in \mathcal{S}$. This can be accomplished by choosing the p_i so that $p_i\sigma_i^2 \approx 0$ for $i \in \mathcal{S}$ and $p_i\sigma_i^2 \approx 1$ for $i \notin \mathcal{S}$.

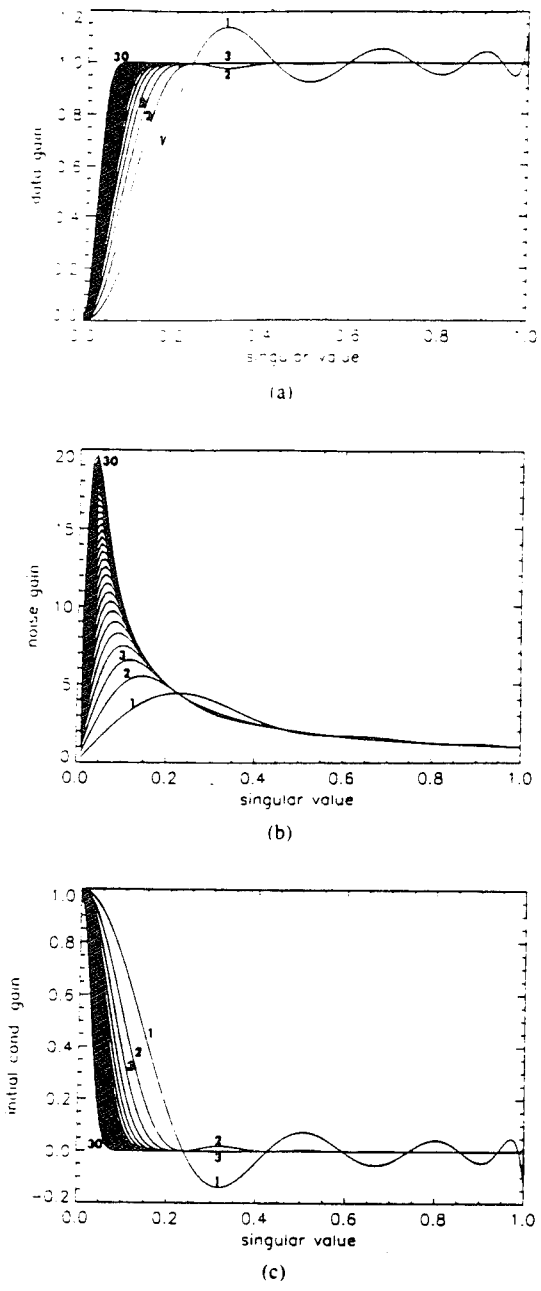


Fig. 2. The convergence properties of (a) data gain $G_{\epsilon}'(\sigma, k)$, (b) noise gain $G_{\epsilon_0}'(\sigma, k)$, and (c) initial condition gain $G_{x_0}'(\sigma, k)$ versus singular-value σ for the first 30 generalized Landweber iterations. The first, second, and third iteration plus the last iteration is labeled.

Then $G_{\epsilon_0}'(\sigma_i, k) \approx 0$ for small values of k and for $i \in \mathbb{S}$. Since $(\epsilon_0, u_i) = 0$ for $i \notin \mathbb{S}$, the second sum of (27) will be almost zero until many iterations have passed, *by which time the rest of the image has converged.* The effect is the same as if the noise had been eliminated, if the iteration is stopped when the image has converged.

Similarly, features in the image can be enhanced relative to other features. To see this, suppose we wish to enhance some features of x in the subspace spanned by $\{v_i, i \in \mathbb{S}'\}$ where $\mathbb{S}' \subset \{1, \cdots, \rho(A)\}$. Then we would like to enhance the terms in the first sum of (27) with $i \in \mathbb{S}'$, or to eliminate the terms with $i \notin \mathbb{S}'$. Choosing the p_i such that $p_i\sigma_i^2 \approx 1$ for $i \in \mathbb{S}'$ and $p_i\sigma_i^2 \approx 0$ for $i \notin \mathbb{S}'$ accomplishes this by making the desired features converge rapidly and the undesired features converge slowly, as desired.

E. Comparison of Computational Requirements

Direct computation of the minimum-norm least-squares solution (8) may require a lot of space for storage of the matrices U and V and a considerable amount of computation time, and it may amplify noise to a great extent. For example, if the system matrix A is 4000×4000, and assuming 4 bytes to store a number in floating-point format, 128 million ($= 4000 \times 4000 \times 4 \times 2$) bytes will be required to store U and V! As for the computation, suppose that the SVD of A is already known after a previous (lengthy) computation, and that A is 97% sparse (3% of the elements of A are nonzero). Direct computation of x^* in (8) is equivalent to about 33 ($\approx 2/0.06$) Landweber iterations if the sparse structure of A is utilized in the Landweber iteration.

In some applications, A becomes sparser as its size is increased. For example, a 150 million-elements system matrix in [6] has about 2 million nonzero elements so that it is 98.7% sparse. The direct computation of x^* in (8) is equivalent to about 75 ($\approx 2/0.026$) Landweber iterations. Hence even apart from storage problems and computing the SVD of A, iterative algorithms become more favorable over direct computation as the size and sparsity of A grows.

The number of forward and backward projections used in one algorithm can be treated as an index indicating how much time the algorithm needs. Generally speaking, the Landweber, ART and MLEM iterations all require one forward and one backward projections for each iteration. Here the operation of multiplying A by a vector is called a "forward projection" and multiplying A^T by a vector is a "backward projection."

To determine the computational load of the generalized Landweber iteration, let $F(\lambda) = a_0 + a_1\lambda + a_2\lambda^2 + a_3\lambda^3$, so that $D = a_0 I + a_1 A^T A + a_2(A^T A)^2 + a_3(A^T A)^3$ where $a_0, \cdots, a_3$ are scalars and coefficients of the polynomial function. Then the generalized Landweber iteration can be written as

$$x^k = x^{k-1} + A^T A(A^T A(a_3 A^T A r^{k-1} + a_2 r^{k-1}) + a_1 r^{k-1}) + a_0 r^{k-1} \qquad (35)$$

where $r^{k-1} = A^T(b - Ax^{k-1})$. Equation (35) requires 4 forward and 4 backward projections. Generalizing, it is clear that using a pth order polynomial function requires $p + 1$ forward and $p + 1$ backward projections, so that one generalized Landweber iteration is approximately equal to $p + 1$ Landweber, ART, or MLEM iterations in terms of computation time.

Refer to the data gains $G_x(\sigma_i, k)$ and $G_x'(\sigma_i, k)$ in (20) and (28), respectively, to get an idea of how many iterations are needed to recover one image component on a particular singular-vector. Table I shows the number of iterations needed to recover 95% of a component on dif-

TABLE I
THE NUMBER OF ITERATIONS NEEDED TO RECOVER 95% OF A COMPONENT ON DIFFERENT SINGULAR-VECTORS

σ	Landweber	Generalized Landweber
0.9	2	2
0.7	5	2
0.5	11	2
0.3	32	2
0.1	299	9
0.09	369	12
0.07	610	19
0.05	1197	38
0.03	3328	106
0.01	29951	951
0.009	36982	1174
0.007	61143	1941
0.005	119951	3804
0.003	332847	10567
0.001	2956472	95188

ferent singular-vectors for the Landweber and generalized Landweber iterations, computed from (20) and (28). The polynomial function (34) is used here in the generalized Landweber iteration. Multiplying the number of iterations for the generalized Landweber iteration in Table I by 7, we can see that the components on singular-vectors with all but the largest three singular-values still converge faster in the generalized Landweber iteration. It is clear that the convergence rate can be accelerated using the generalized Landweber iteration, even though each iteration takes more time to compute.

IV. The Multigrid Method

The purpose of applying a multigrid method to image reconstruction is to accelerate the iterative reconstruction by first using a coarse-grid iteration to provide an initial condition for the more computation—demanding fine-grid iteration. The idea from [22] is that low-frequency components of an image can be reconstructed with less effort using the coarse-grid iteration; then applying the fine-grid iteration results in an efficient reconstruction of high-frequency components.

Ranganath *et al.* [22] used a multigrid method to accelerate the convergence rate of the MLEM algorithm. It was shown in [22] that this multigrid-MLEM approach reduced the computation time for reconstructing a 128 × 128 noisy Shepp–Logan phantom, using a sum of the squared errors criterion. However, the convergence behavior of the multigrid method for image reconstruction in general, was not considered in [22].

In Section V, we will investigate the convergence properties of the multigrid method, using an interpolation technique in which each coarse-grid pixel is evenly divided or distributed into its four neighboring fine-grid pixels. This interpolation is identical to the zero-order interpolation [27], except for a multiplicative constant of one quarter which we include to preserve the number of total counts. The multigrid implementations of the Landweber, generalized Landweber, ART without relaxation [2], and MLEM [5] algorithms are all investigated. Three noise-free images with different frequency contents are used in this study.

Our multigrid implementation is set up as follows. A coarse-grid is defined where the center of a coarse-grid pixel is located at the center of the four neighboring fine-grid pixels. Then a coarse-grid system matrix associated with the coarse-grid is computed and stored. Note that the set of projections (elements of b) for the coarse-grid geometry usually does not coincide with the projections of the fine-grid geometry.

In our multigrid method, the projection data are first derived from a noiseless projection of a synthetic image through the fine-grid system. A subset of projection data which corresponds to the set of projections of the coarse-grid system is chosen as the simulated projections from the synthetic image for the coarse grid system. For conservation of total counts, the total counts of the coarse-grid projection data will be normalized such that the total counts of the coarse-grid system will be the same as that of the fine-grid system. Since we are investigating to what extent the multigrid method can contribute to the image reconstruction, we use directly the minimum-norm least-squares solution of the coarse-grid system, which is equivalent to the convergent solution of the coarse-grid iterations (Landweber, generalized Landweber and ART) with zero initial condition. Then the interpolation is applied on the coarse-grid convergent solution to make a fine-grid image. A positivity constraint and another conservation of total counts are applied on the fine-grid image to make up a fine-grid initial image. Finally, the fine-grid iteration with this fine-grid initial condition is executed 1000 times for investigation of its convergence behavior.

V. Numerical Results, Discussion, and Comparison

A. Experiments on Small Scale Systems

Two hypothetical small-scale PET systems are used in this section. Each has a detector ring of radius 1, and a centered object support of 1.2 × 1.2. The detector rings of PET-1 and PET-2 are equally divided into 26 and 16 detectors, respectively; it is assumed that there is no gap between any two adjacent detectors. For applying the multigrid method, an object will be pixelated as either 144 (= 12 × 12) fine-grid pixels or 36 (= 6 × 6) coarse-grid pixels.

A tube is defined by any two different detectors [5]; some of the tubes defined may not cover any pixel of the image at all. Theoretically, there are 325 (= 26 × 25/2) and 120 = (16 × 15/2) different tubes for PET-1 and PET-2, respectively. However, in order to reduce computation time, the number of different tubes or projections is defined as the number of tubes which have nontrivial intersections with at least one pixel.

We assume the conditional probability $P(t/i)$ of an annihilation occurring at pixel i and its consequent coinci-

dence event being detected by tube t is proportional to the angle of "looking" from pixel i into tube t, which is

$$P(t/i) = \frac{\text{angle in radians from pixel } i \text{ looking into tube } t}{2\pi}. \tag{36}$$

A normalization [5] makes $\Sigma_{t=1}^{t^*} P(t/i) = 1$ where t^* is the total number of tubes. The system matrix A is defined by having $\langle A \rangle_{t,i} = P(t/i)$ after normalization.

Two fine-grid system matrices A_1 (dimension 195 × 144) and A_2 (84 × 144), both with 144 pixels, are defined for PET-1 and PET-2, respectively, and two coarse-grid system matrices A_1' (dimension 175 × 36) and A_2' (72 × 36), both with 36 pixels, are also defined for PET-1 and PET-2, respectively. SVD analysis shows there are 144 and 84 nonzero singular-values for matrices A_1 and A_2, respectively, which indicate that $N(A_1)$ is empty, i.e., $N(A_1) = \phi$, and $N(A_2)$ has dimension 60 = (144 − 84) [24].

Fig. 3 shows three synthesized phantoms or images, which are called image-1, image-2, and image-3. Image-1 primarily contains low-frequency components, while image-2 and image-3 are explicitly chosen to illustrate the importance of the "local smoothness" property defined below (image-2 does not have this property; image-3 does). Three different initial conditions are investigated: 1) *zero* initial condition, in which every pixel is zero, 2) *average* initial condition, in which every pixel is initialized to be the same value, which is total counts divided by the number of pixels, and 3) *multigrid* initial condition, which comes from the interpolation of the minimum-norm least-squares solution of the coarse-grid system. Note that the zero initial condition cannot be used for the MLEM iteration [5].

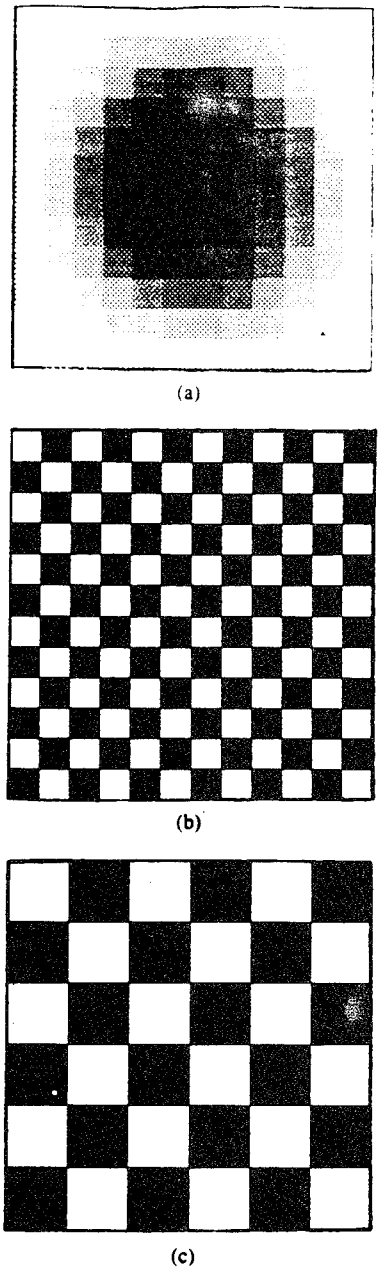


Fig. 3. (a) Image-1 (b) image-2, and (c) image-3. Each image is scaled to have black indicating the biggest magnitude, and white indicating the smallest magntiude. Image-1 has no high-frequency components, image-2 does not have the local smoothness property, and image-3 does have the local smoothness property.

B. Results for Small-Scale Systems

Figs. 4 and 5 show the performance for the Landweber, generalized Landweber, ART, and MLEM iterations with the three images in Fig. 3 and three types of initial conditions (zero, average, multigrid) for PET-1 and PET-2, respectively. Each curve shows results up to 1000 iterations. In Fig. 5, there are also some horizontal lines which represent the results of setting $p_i = 1/\sigma_i^2$ in (27). Any one of these represents a least-squares solution made up of 1) the minimum-norm least-squares solution (8), which depends on the original image vector x, and 2) the null space component $x_0^{\perp}$, which is the projection of the initial condition (zero, average, or multigrid) x_0 onto $N(A_2)$. These lines serve as lower bounds on Euclidean distance for the least-squares type algorithms: ART (see corollary 9 of Tanabe [23]); Landweber [see (19)]; and generalized Landweber [see (27)]. The bounds are inapplicable for the MLEM algorithm. Note that the MLEM algorithm with average initial condition breaks some of these bounds in Fig. 5(a).

In Fig. 5(c) [image-3 in PET-2], the multigrid implementations of all algorithms break the lower bounds with zero and average initial conditions! In fact, the coarse-grid iteration has recovered some image features lying in the null space $N(A_2)$ of the fine grid.

Since the coarse-grid and fine-grid system geometries differ, the coarse-grid solution after interpolation may include components lying in the null space of the fine-grid system matrix. In this example, the result of the coarse-grid iteration furnishes extra information. The coarse-grid iteration basically estimates the value of a coarse-grid pixel representing its four neighboring fine-grid pixels, which have the same magnitude in image-3, and the interpolation does not violate any edge or boundary of this particular image. Therefore, some high-frequency com-

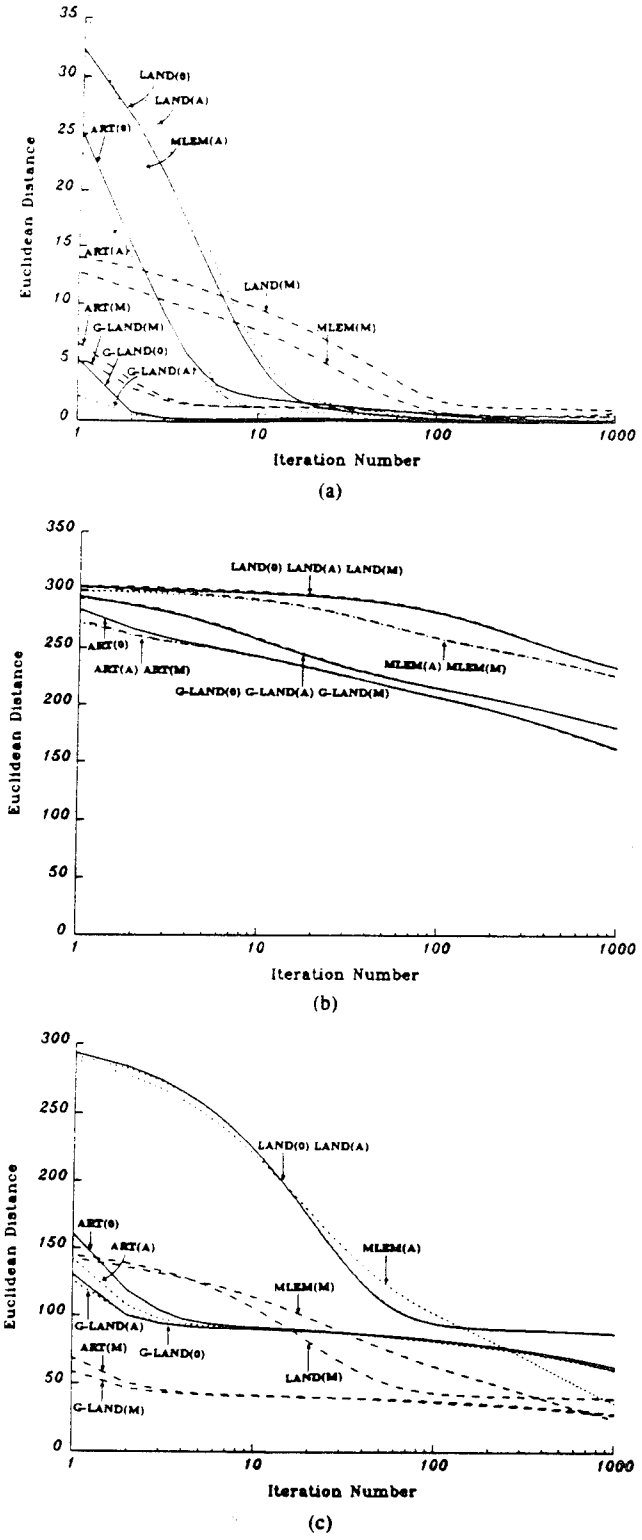


Fig. 4. Euclidean distance between reconstructed and actual images versus iteration number in PET-1 for (a) image-1, (b) image-2, and (c) image-3 for different algorithms. LAND(0): Landweber iterations with zero initial condition, G-LAND(A): generalized Landweber iteration with average initial condition, LB(M): Euclidean distance lower bound for multigrid initial condition, etc.

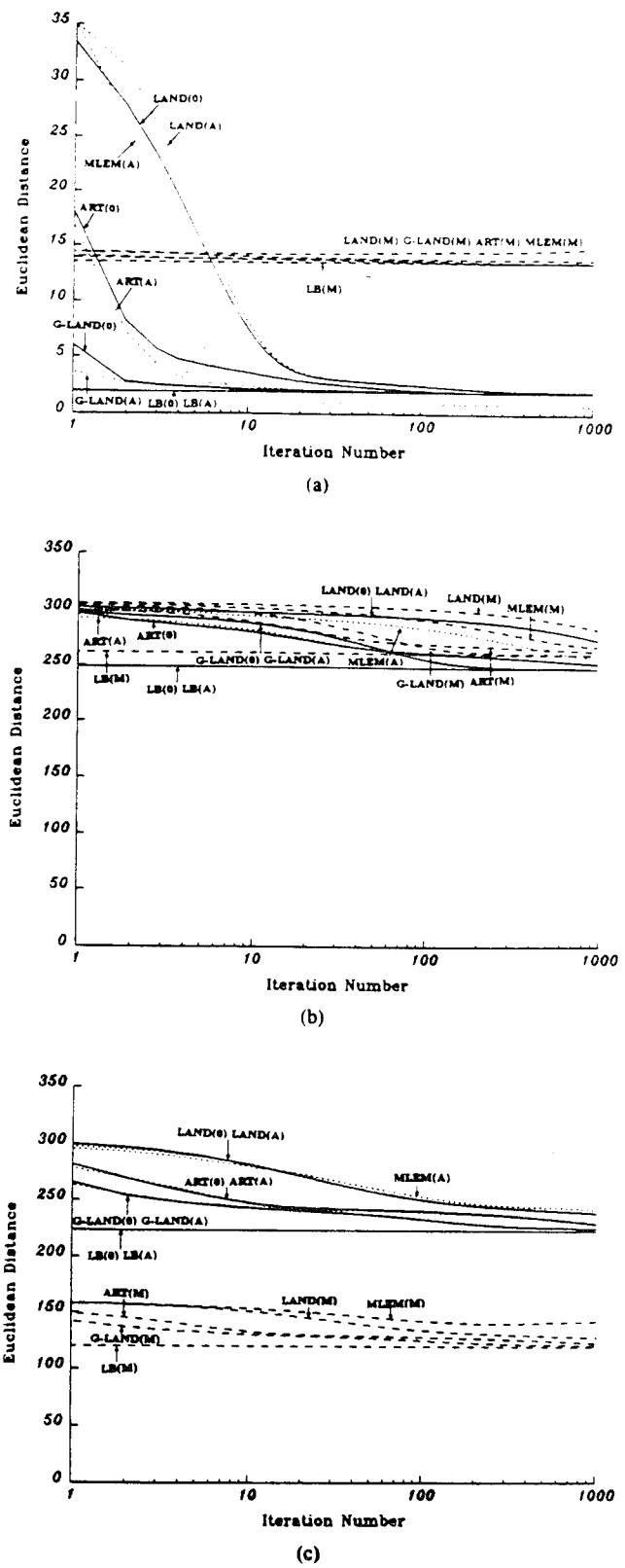


Fig. 5. Euclidean distance between reconstructed and actual images versus iteration number in PET-2 for (a) image-1, (b) image-2, and (c) image-3 for different algorithms.

ponents may be picked up from this multigrid procedure for image-3.

We now define a *local smoothness property:* the values of neighboring four fine-grid pixels to be grouped together as a coarse-grid pixel are close to each other. Image-3 has this property, and the convergence of its high-frequency components is significantly accelerated using the multigrid implementation. The improvement in qual-

ity of the reconstructed image for image-3 in Fig. 5(c) can be seen by comparing Fig. 6(a), the convergent result of using zero initial condition [by setting $p_i = 1/\sigma_i^2$ in (27)], and Fig. 6(b), the convergent result of using the multigrid initial condition (also by setting $p_i = 1/\sigma_i^2$). Fig. 6(b) more closely resembles the original image Fig. 3(c).

C. Discussion of Results for Small-Scale Systems

We now summarize the results in Figs. 4 and 5, and discuss them with the aid of decompositions or representations of the images on the singular-vectors v_i of both A_1 and A_2. Since the computation in one generalized Landweber iteration with the polynomial function $F(\lambda)$ in (34) is almost equivalent to 7 Landweber, ART, or MLEM iterations, the abscissae indicating iteration number in Figs. 4 and 5 do not show actual computation time for the generalized Landweber iteration; since the abscissae have logarithmic scales, the generalized Landweber curves may simply be shifted to the right to reflect the actual computation required. The discussion below is based on amount of computation, not number of iterations.

For zero or average initial condition (see Figs. 4 and 5).

1) The Landweber iteration is usually the slowest of the four methods.

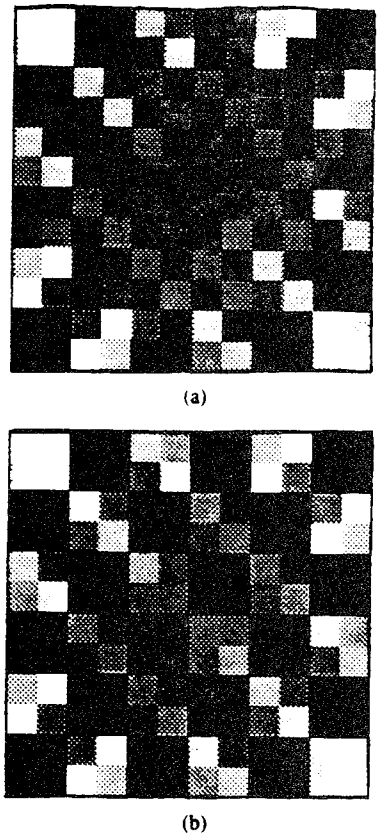
2) The MLEM iteration is usually slower than ART, but faster than the Landweber iteration. However, in some cases, such as image-1 and image-3 in PET-1 [see Fig. 4(a) and (c)], the MLEM iterations with average initial condition [in Fig. 4(a)] and with average and multigrid initial conditions [in Fig. 4(c)] converge faster than the generalized Landweber and ART iterations, and the MLEM iteration with average initial condition in the case of image-1 in PET-2 [see Fig. 5(a)] can even break the least-squares lower bounds.

In order to understand the convergence behavior in Figs. 4 and 5, the three images in Fig. 3 are decomposed into components on the singular-vectors v_i of both systems. The decompositions for PET-1 and PET-2 are shown in Figs. 7 and 8, respectively. There are 144 singular-vectors, ordered by the magnitudes of the singular-values from right to left in the abscissa, with index 144 for the singular-vector with the largest singular-value, and with index 1 for the singular-vector with the smallest singular-value. If the image has high frequency components, i.e., nonzero and large components on the left, the iteration tends to take longer to converge [see also Fig. 1(a) and Fig. 2(a)].

For the multigrid initialization (see Figs. 4 and 5).

1) In Fig. 5(a), the reconstructed image hardly changes in the course of iteration for image-1, its error lying just above the least-squares bound. This suggests that the initial condition may be very close to a least-squares solution.

2) In Fig. 4(c) and Fig. 5(c), the multigrid iteration accelerates the convergence rate dramatically for image-3 in both systems;



(a)

(b)

Fig. 6. Reconstructed images using (a) zero and (b) multigrid initial conditions on image-3 in PET-2. Each image is scaled to have black indicating the biggest magnitude, and white indicating the smallest magnitude.

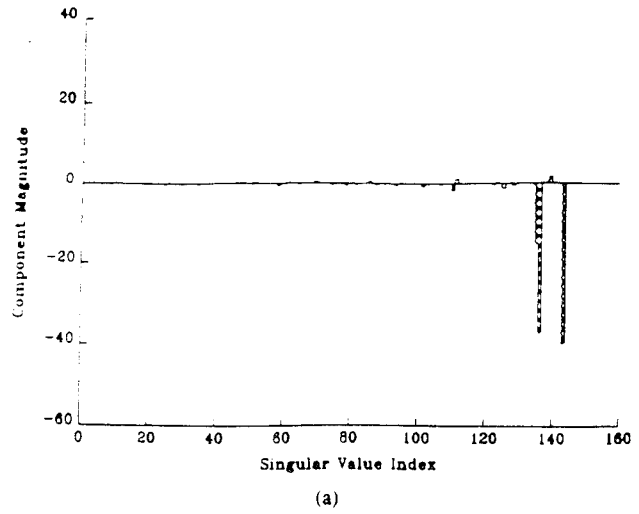
3) In Fig. 4(a), (b) and Fig. 5(b), the multigrid method does not show any significant improvement over single-grid algorithms for image-1 or image-2.

D. Comparison of Results for Small-Scale Systems

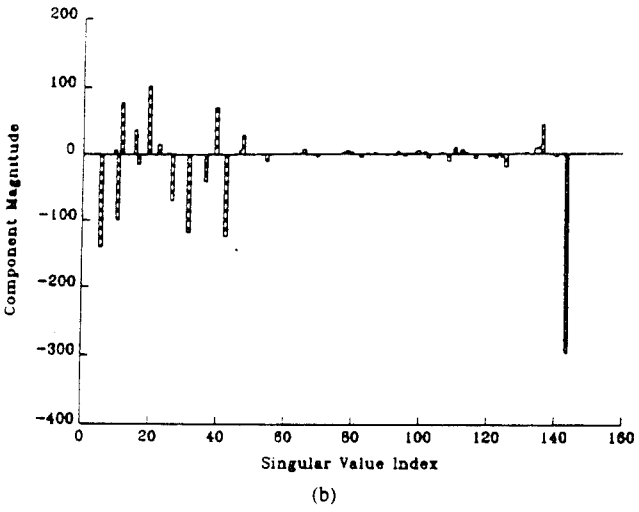
With the aid of the decompositions of the multigrid and average initial conditions for PET-1 and PET-2 shown in Figs. 9 and 10, respectively, we can make the following observations.

1) Refer to the decompositions of average initial conditions in PET-1 and PET-2 in Fig. 9(d)–(f) and Fig. 10(d)–(f), respectively. Since the average initial conditions only contain low-frequency components (on the singular-vectors with biggest singular-values), and these components may be recovered after the first one or two iterations (see Table I), using the average initial condition may not speed up the convergence.

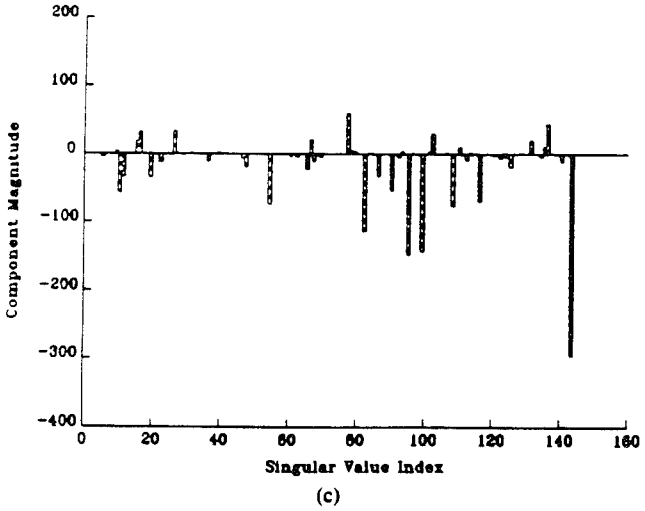
2) Compare the decomposition of image-1 in PET-1 [see Fig. 7(a)] to the decomposition of the multigrid initial condition for image-1 in PET-1 or its corresponding multigrid decomposition [see Fig. 9(a)]. The multigrid initial condition not only contains some low-frequency components, but also introduces some high-frequency "noisy" components which will take many iterations to be corrected. In Fig. 4(a), the multigrid Landweber and multigrid MLEM implementations, after several iterations, are slower than their counterparts with zero or aver-



(a)

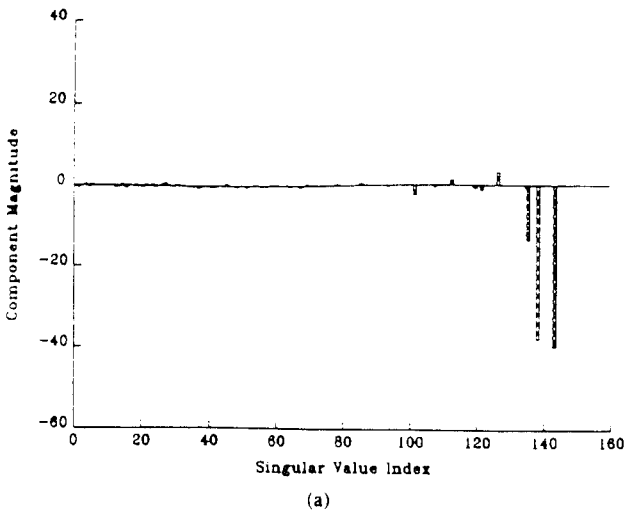


(b)

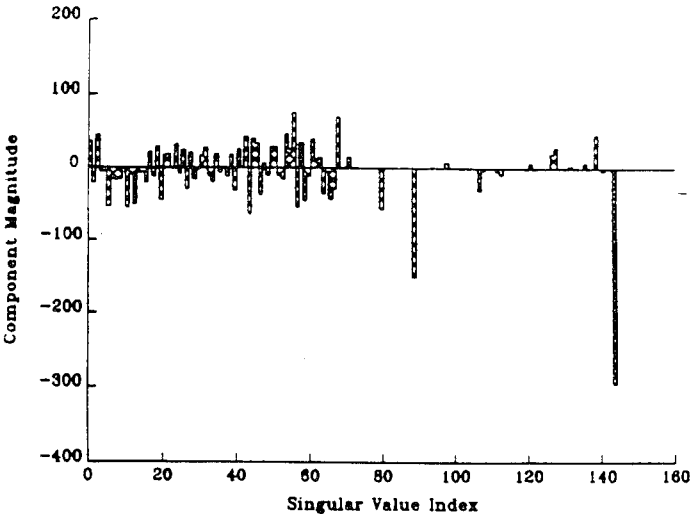


(c)

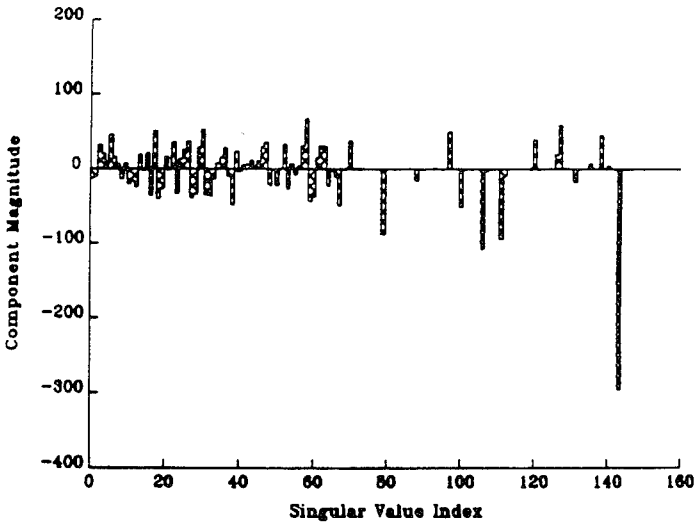
Fig. 7. The decompositions of (a) image-1, (b) image-2, and (c) image-3 on the singular-vectors v_i of A_1.



(a)



(b)



(c)

Fig. 8. The decompositions of (a) image-1, (b) image-2, and (c) image-3 on the singular-vectors v_i of A_2.

age initial condition, and the multigrid generalized Landweber iteration is slower than its counterpart with zero or average initial condition.

3) Compare the decomposition of image-1 in PET-2 [see Fig. 8(a)] to its corresponding multigrid decomposition [see Fig. 10(a)]. The multigrid initial condition, which is close to a least-squares solution, has a component similar to the minimum-norm least-squares solution

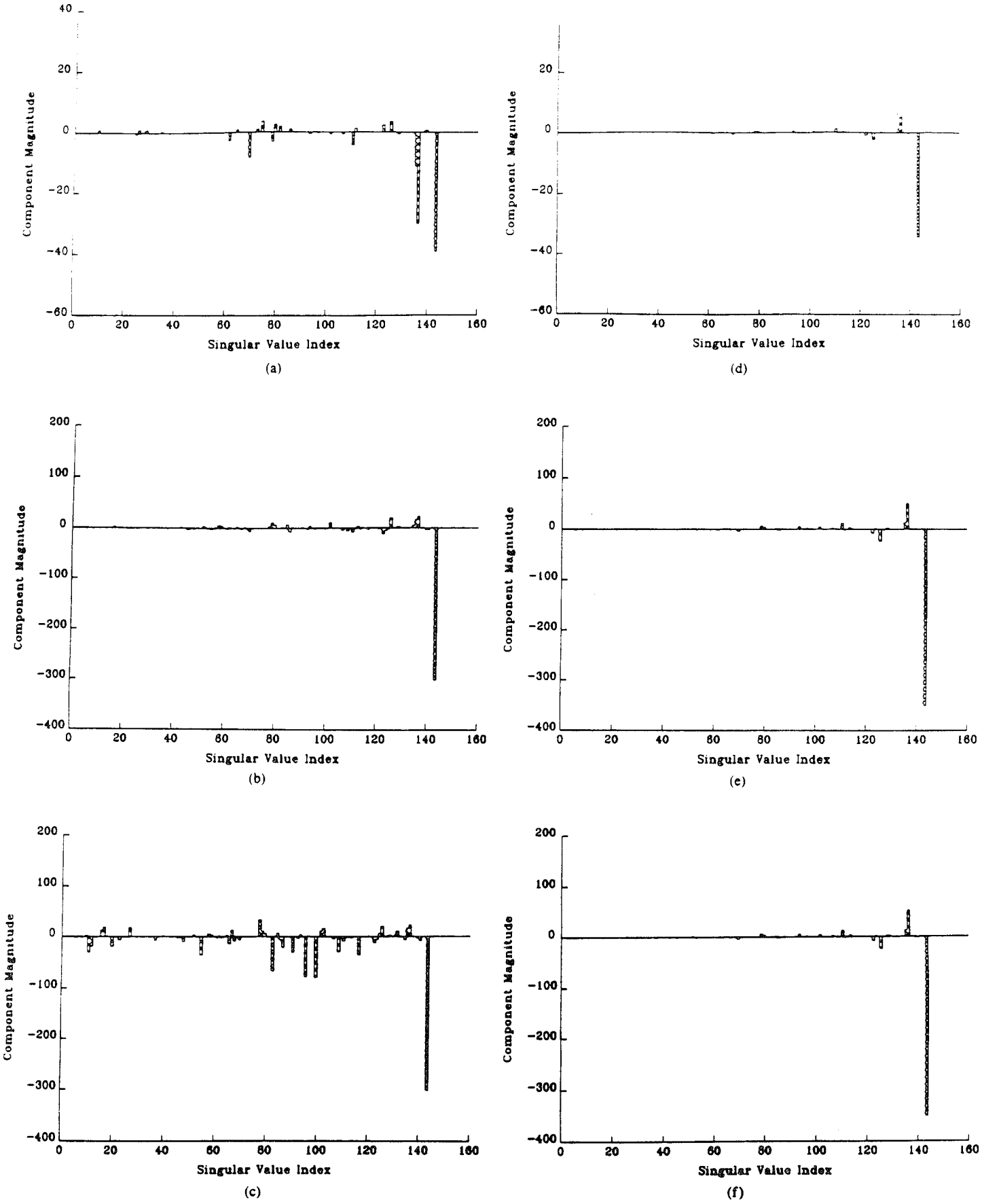
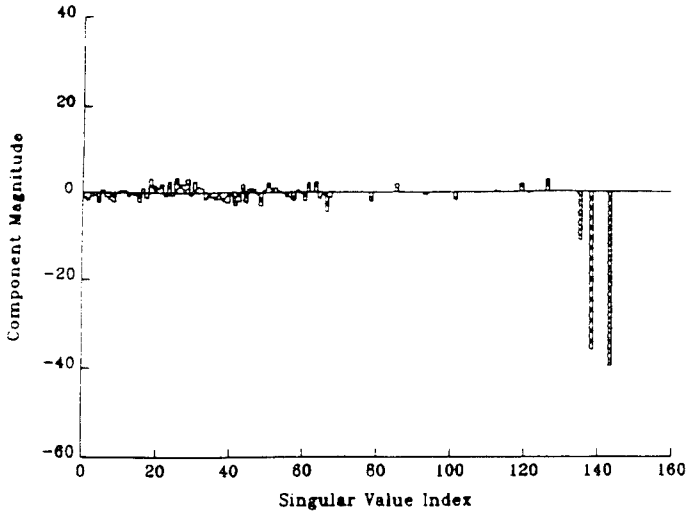
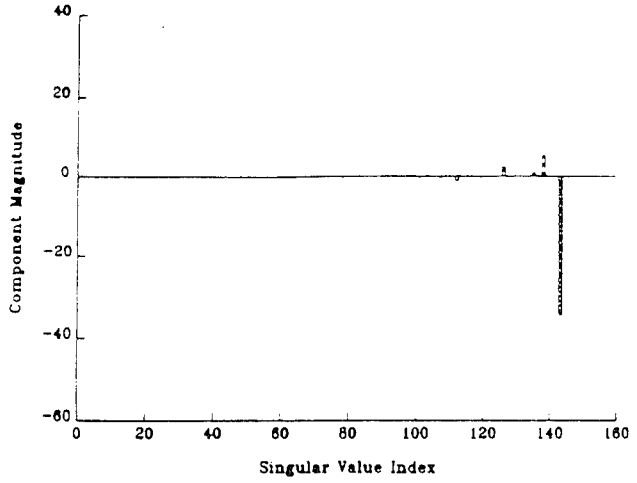


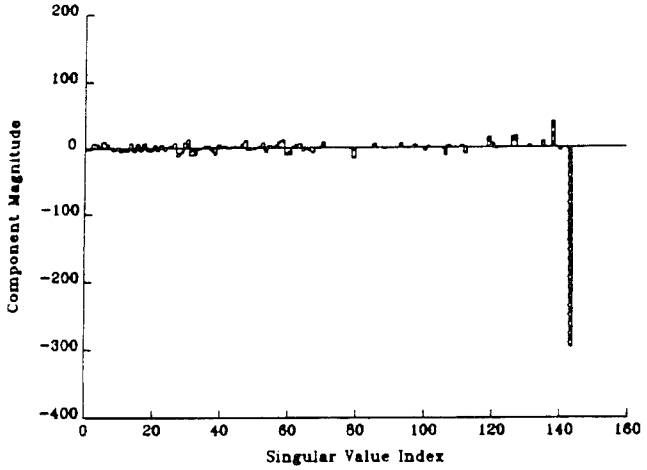
Fig. 9. The decompositions of the multigrid initial conditions of (a) image-1, (b) image-2, and (c) image-3, and the decompositions of the average initial conditions of (d) image-1, (e) image-2, and (f) image-3 on the singular-vectors v_i of A_1.



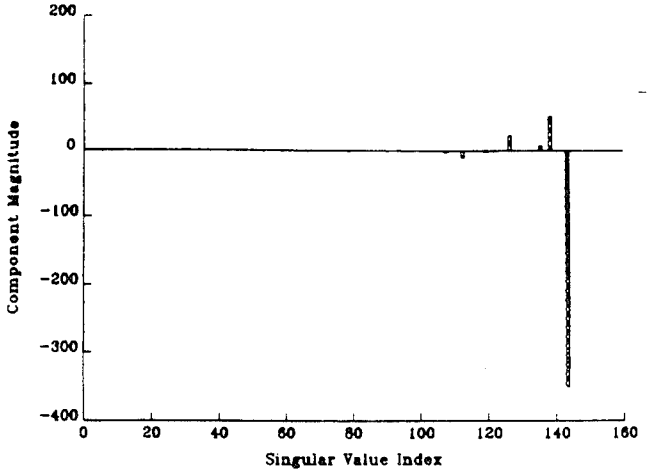
(a)



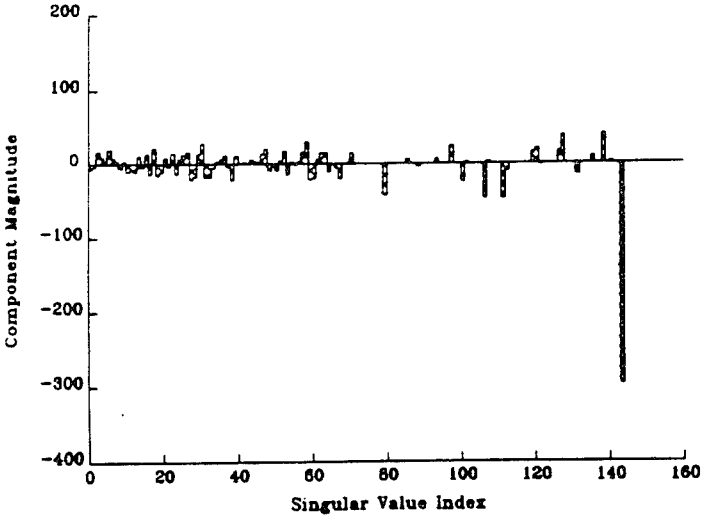
(d)



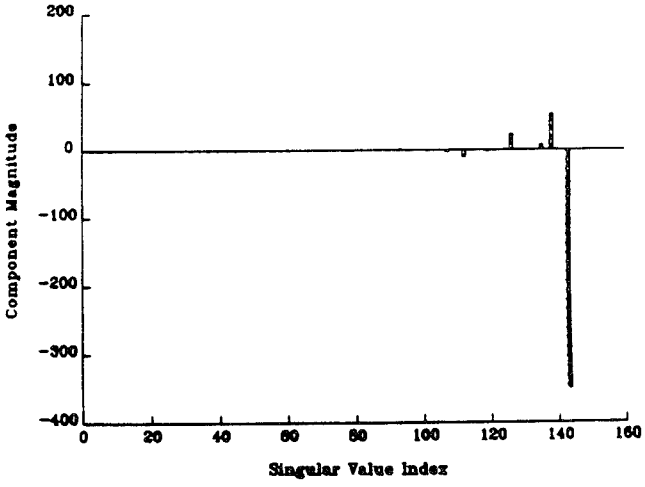
(b)



(e)



(c)



(f)

Fig. 10. The decompositions of the multigrid initial conditions of (a) image-1, (b) image-2, and (c) image-3, and the decompositions of the average initial conditions of (d) image-1, (e) image-2, and (f) image-3 on the singular-vectors v_i of A_1.

represented by the elements with indices from 61 to 144, and also has a component represented by the elements with indices from 1 to 60 in $N(A_2)$. This moves each reconstructed image x^k a distance away from the true solution [see Fig. 5(a)].

4) Compare the decomposition of image-2 in PET-1 [see Fig. 7(b)] to its corresponding multigrid decomposition [see Fig. 9(b)], and the decomposition of image-2 in PET-2 [see Fig. 8(b)] with its corresponding multigrid decomposition [see Fig. 10(b)]. The multigrid initial conditions do not recover most of the components, and do not help to accelerate the convergence rate for image-2 in both systems. The reason for this is as follows. Every coarse-grid pixel is split into 4 equivalent parts to initialize the fine-grid iteration. However, since image-2 does not have the local smoothness property defined above, this fine-grid initialization does not resemble the actual image.

5) Compare the decomposition of image-3 in PET-1 [see Fig. 7(c)] to its corresponding multigrid decomposition [see Fig. 9(c)], and the decomposition of image-3 in PET-2 [see Fig. 8(c)] with its corresponding multigrid decomposition [see Fig. 10(c)]. The multigrid initial conditions not only contain some low-frequency components for image-3 in both systems, but also contain some high-frequency components in PET-1 and some components in the null space $N(A_2)$ of PET-2. Therefore using the multigrid method can greatly accelerate the convergence rate for image-3, which does have the local smoothness property. Also since $N(A_2)$ is not empty, it is impossible for the iterations with zero initial condition to catch up to the multigrid iterations for image-3 in PET-2.

E. Conclusions from Results for Small-Scale Systems

The decompositions in Figs. 7–10 show that the multigrid method accelerates the convergence rate for high-frequency components if the image has the local smoothness property, as image-3 does.

The multigrid method does not improve the convergence rate when the image has no high-frequency components, as in image-1, or does not have the local smoothness property, as in image-2. Note that *any* image will have the local smoothness property if enough pixels are used to describe it; conversely, the multigrid method could be used to identify images described by too many pixels.

The advantage of using the multigrid method, from our simulation results, is its capability of recovering high-frequency components more quickly, if an image has the local smoothness property. It is true that if the coarse-grid iteration recovers some low-frequency components of the image for the subsequent fine-grid iteration, then the fine-grid iteration will perform better in the reconstruction of the low-frequency components in its first several iterations. However, the fine-grid iteration itself may be very capable of recovering low-frequency components, and the benefit of using the coarse-grid iteration to effectively recover low frequencies in the multigrid method may soon disappear after the first several fine-grid iterations. Then the single-grid iteration will be as good as the multigrid iteration. The use of two different grid geometries (coarse and fine) may result in the recovery of elements in the null space of the fine-grid system matrix. In terms of frequencies, this can be interpreted by noting that null space components can be viewed as infinitely high-frequency components (i.e., zero singular-values), which can never be recovered by using the fine-grid iteration with zero initial condition.

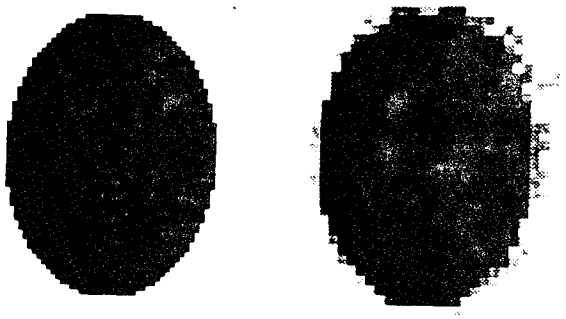
In some cases, the coarse-grid iteration may be unable to recover high-frequency components, and it may even introduce some incorrect high-frequency components! Then the multigrid method should show no sign of being able to accelerate the convergence rate after the first several iterations. This may help to explain why the paths of convergence for the multigrid Landweber and multigrid MLEM iterations in Fig. 4(a) start better than the single-grid implementations in the first several iterations, and then become worse after about the ninth iteration in the Landweber and the fourth in the MLEM.

We refer to this as the "starting good but ending bad" phenomenon, which is due to a nonproper initial condition setting for the fine-grid iteration. This may explain a result in [14], in which using the image from FBP to initialize the conjugate gradient iteration [14] did *not* accelerate the convergence rate of the iteration (see [14, Fig. 4]). Despite the good starting point, the conjugate gradient method showed this phenomenon after the 75th iteration. This may be because the FBP initial condition might have introduced some high-frequency "noisy" components into the reconstructed image, which dominated the image in the iteration after the low-frequency components had been recovered in the first 75 iterations.

F. Experiments on a Large Scale System

A geometry similar to the one in [5], with 128 detectors and a square array of 4096 ($= 64 \times 64$) fine-grid pixels circumscribed by the detector ring, was used as a more realistic large-scale system to verify the conclusions from the previous small scale systems. The object support was assumed to be in the inscribed circle of the square array, which accounts for 3228 fine-grid pixels or 812 coarse-grid pixels, determined by counting the center positions of fine or coarse grids inside the inscribed circle. Again neglecting projections that did not intersect any pixel in the object support, we found that the corresponding fine-grid system geometry was modeled by a system matrix A_3 of 4160×3228, and the coarse-grid system by A_3' of 4152×812.

The Shepp–Logan phantom [12] shown in Fig. 11(a) is used as the object. This phantom can be seen to satisfy the local smoothness property, except at its boundaries. The multigrid procedure was identical to that used on the small-scale systems, with the exception that the coarse-grid convergent solution was obtained using a conjugate gradient iteration [14], [25], to ensure convergence after a finite number of iterations [25] (812 here). The result of



(a) (b)

Fig. 11. (a) Shepp–Logan phantom and (b) multigrid initial condition. Each image is scaled to have black indicating the biggest magnitude, and white indicating the smallest magnitude.

the coarse-grid iteration was investigated. It was found that the reconstructed image of the coarse-grid iteration did not change much (in terms of Euclidean distance between two consecutive images) after about 16 iterations, which are almost equivalent in computation to four fine-grid iterations. Note one single conjugate-gradient iteration needs two forward and two backward projections [14]. So the four fine-grid conjugate-gradient iterations are about equal to one fine-grid generalized Landweber iteration, or eight fine-grid Landweber, ART, or MLEM iterations. The convergent image after interpolation is shown in Fig. 11(b), which was used as the multigrid initial condition for all fine-grid iterations.

The simulation results are shown in Fig. 12. The Landweber, ART, and MLEM iterations all benefit from the multigrid implementation, even with eight extra for Landweber, ART, and MLEM fine-grid iterations included to account for the coarse-grid iterations. Only the generalized Landweber iteration does not improve its convergence behavior. The reason is that the multigrid initial condition contains only the frequency components with singular values of $\sigma > 0.2$, and these will disappear after only three generalized Landweber iterations [see Fig. 2(c)]. If the phantom is represented as an image of 128×128 pixels, as in [22], then the local-smoothness property will become more prominent and the convergence rate may be accelerated even more.

One interesting point about ART is that we do not get quite the same convergence behavior as we saw in the small scale systems PET-1 and PET-2. Here [see Fig. 12(c)], the ART iteration with zero initial condition converges considerably slower than the same iteration with average or multigrid initial condition in the first 40 iterations. The reason may be that the average initial condition (i.e., every pixel is positive) is geometrically closer to the true object (the Shepp–Logan phantom, whose pixels are all positive) than the zero initial condition. This makes the iteration much faster.

G. Comparison to the Results of Ranganath

In [22], Ranganath *et al.* proposed a multigrid MLEM algorithm. They claimed that "*the low-frequency components of the solution are effectively approximated and recovered on the coarse grids while the high-frequency components are recovered on the fine grids.*" The first part of the statement about one of the capabilities of the coarse-grid iteration is correct. However, the potential capabilities of the multigrid method to recover high-frequency components in the case of local smoothness, and to introduce incorrect high-frequency components, were not mentioned. The second part of the statement about the role played by the fine-grid iteration is also correct. However, this role should not be emphasized in using the multigrid method for the following three reasons.

1) For the Landweber and generalized Landweber iterations, the third sums in (19) and (27) show that the initial condition dies out geometrically [this is illustrated in Fig. 1(c) and Fig. 2(c)]. This suggests that the low-frequency components recovered by the coarse-grid iteration do not remain unaltered during the fine-grid iteration. Hence the fine-grid iteration must do more than reconstruct high-frequency components.

2) The convergence rate of the fine-grid iteration in recovering high-frequency components is not good, unless the local smoothness property also holds (as it might be in the example used in [22]).

3) Because of the nonuniform convergence property noted in Section III, it is likely that, for images without high-frequency components, after the first several iterations the single-grid iteration may be as good as the multigrid iteration, or even better if some incorrect high frequency components were introduced by the coarse-grid iteration.

In interpreting the results of [22] in light of the guidelines proposed in Section V-E, the following points should be noted.

1) The original image was represented by 128×128 pixels, which benefits the multigrid method, since the fine sampling induces the local smoothness property.

2) The number of iterations used was small: "four" fine-grid iterations. This again shows the multigrid method in its best light. Since it has received some low-frequency components from the coarse-grid iteration, the subsequent fine-grid iteration will seem to have effectively recovered more low-frequency components in its first several iterations than the single-grid iteration. The fine-grid iteration may in fact have not done much for the high-frequency components in these first several iterations.

The benefit of applying the MLEM multigrid approach on the Shepp–Logan phantom in the first several iterations can also be seen in Fig. 12(d). Thus the results of [22] seem to be in agreement with the guidelines for the multigrid method. However, this paper has treated several different images, and multigrid implementations of several different algorithms.

VI. Summary

A numerical study of multigrid implementations of several iterative image reconstruction algorithms has been

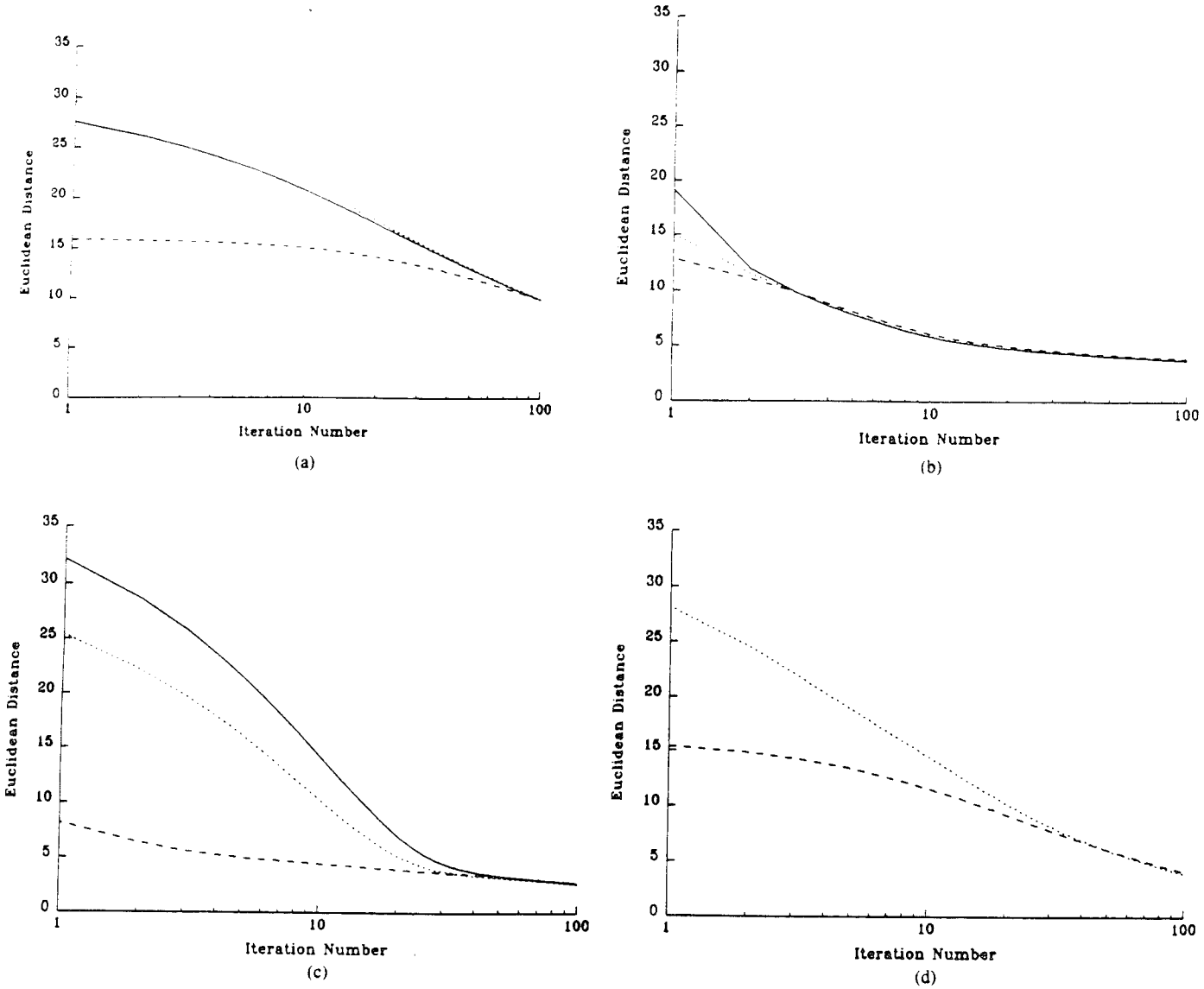


Fig. 12. Euclidean distance between reconstructed and actual images versus iteration number for (a) Landweber, (b) generalized Landweber, (c) ART, and (d) MLEM. Solid line is for zero initial condition, dotted line for average, and dashed line for multigrid.

presented. The generalized Landweber iterations was shown to have a faster convergence rate than the Landweber iteration if a proper shaping matrix is used. And the generalized Landweber iteration has several advantages over ART, including parallelizability, control over convergence rate, and ability to filter the data or image as a part of the iteration. The effects of noise and initial condition on the convergence rate of the generalized Landweber iteration were also studied.

The multigrid implementation was found to accelerate the convergence rate of high-frequency components of the image when the image possessed the local smoothness property. In other cases it was unhelpful, and may even slow down the convergence rate. Results of some other papers were interpreted in light of the conclusions drawn here. Unresolved issues include determination of sampling rates that will ensure the local smoothness property, other ways of initializing the iterations, and incorporation of a more general nonlinear D operator (e.g., a nonnegativity constraint) in the generalized Landweber iteration. We believe this paper has presented some valuable insight into the convergence behavior of various implementations of various iterative algorithms.

Acknowledgment

The authors wish to thank Dr. W. L. Rogers and N. H. Clinthorne of the Division of Nuclear Medicine of the University of Michigan for many helpful discussions and for bringing the generalized Landweber iteration to their attention.

References

[1] E. Tanaka, "Recent progress on single photon and positron emission tomography-from detectors to algorithms," *IEEE Trans. Nucl. Sci.*, vol. NS-34, pp. 313-320, Feb. 1987.

[2] Y. Censor, "Finite series-expansion reconstruction methods," *Proc. IEEE*, vol. 71, pp. 409–419, Mar. 1983.

[3] E. Tanaka, "A fast reconstruction algorithm for stationary positron emission tomography based on a modified EM algorithm," *IEEE Trans. Med. Imaging*, vol. MI-6, pp. 98–105, June 1987.

[4] H. A. Klotz, Jr. and D. L. Snyder, "A hardware architecture using finite-field arithmetic for computing maximum-likelihood estimates in emission tomography," *IEEE Trans. Med. Imaging*, vol. MI-7, pp. 279–290, Dec. 1988.

[5] L. A. Shepp and Y. Vardi, "Maximum likelihood reconstruction for emission tomography," *IEEE Trans. Med. Imaging*, vol. MI-1, pp. 113–122, Oct. 1982.

[6] L. Kaufman, "Implementing and accelerating the EM algorithm for positron emission tomography," *IEEE Trans. Med. Imaging*, vol. MI-6, pp. 37–51, Mar. 1987.

[7] E. Veklerov and J. Llacer, "Stopping rule for the MLE algorithm based on statistical hypothesis testing," *IEEE Trans. Med. Imaging*, vol. MI-6, pp. 313–319, Dec. 1987.

[8] J. Llacer and E. Veklerov, "Feasible image and practical stopping rules for iterative algorithms in emission tomography," *IEEE Trans. Med. Imaging*, vol. MI-8, pp. 186–193, June 1988.

[9] J. A. Stamos, W. L. Rogers, N. H. Clinthorne, and K. F. Koral, "Spectral differences as a means for evaluating performance of iterative reconstruction algorithms," (Abstract) *J. Nucl. Med.*, vol. 27, p. 896, 1986.

[10] W. L. Rogers and N. H. Clinthorne, "Fundamental elements of single photon emission computerized tomography," in *Radiology: Diagnosis-Imaging-Intervention*. Philadelphia, PA: Lippincott, 1988, vol. 1, ch. 23, pp. 1–8.

[11] W. L. Rogers and N. H. Clinthorne, "Single photon computed tomography (SPECT)," in *Nuclear Medical Physics*, L. E. Williams Ed. Florida: CRC Press, 1987, vol. 3, ch. 1, pp. 1–48.

[12] A. C. Kak and M. Slaney, *Principles of Computerized Tomographic Imaging*. New York: IEEE Press, 1988.

[13] C. K. Rushforth, "Signal restoration, functional analysis, and Fredholm integral equations of the first kind," in *Image Recovery: Theory and Applications*, H. Stark, Ed. New York: Academic, 1987, ch. 1, pp. 1–27.

[14] S. Kawata and O. Nalcioglu, "Constrained iterative reconstruction by the conjugate gradient method," *IEEE Trans. Med. Imaging*, vol. MI-4, pp. 65–71, June 1985.

[15] G. Demoment, "Image reconstruction and restoration: Overview of common estimation structures and problems," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-37, pp. 2024–2036, Dec. 1989.

[16] L. Landweber, "An iterative formula for Fredholm integral equations of the first kind," *Amer. J. Math.*, vol. 73, pp. 615–624, 1951.

[17] O. N. Strand, "Theory and methods related to the singular-function expansion and Landweber's iteration for integral equations of the first kind," *Siam J. Numer. Anal.*, vol. 11, no. 4, pp. 798–825, Sept. 1974.

[18] H. J. Trussell and M. R. Civanlar, "The Landweber iteration and projection onto convex sets," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, pp. 1632–1634, Dec. 1985.

[19] H. J. Trussell, "Convergence criteria for iterative restoration methods," *IEEE Trans. Acoust., Speech, Signal Processing.*, vol. ASSP-31, pp. 129–136, Feb. 1983.

[20] P. A. Jansson, *Deconvolution with Applications in Spectroscopy*. New York: Academic, 1984.

[21] B. R. Frieden, "Image Enhancement and Restoration," in *Picture Processing and Digital Filtering*, T. S. Huang, Ed. New York: Academic, 1979, ch. 5, pp. 177–248.

[22] M. V. Ranganath, A. P. Dhawan, and N. Mullani, "A multigrid expectation maximization reconstruction algorithm for positron tomography," *IEEE Trans. Med. Imaging*, vol. 7, pp. 273–278, Dec. 1988.

[23] K. Tanabe, "Projection method for solving a singular system of linear equations and its applications," *Numerische Mathematik*, vol. 17, pp. 203–214, 1971.

[24] B. Noble and J. W. Daniel, *Applied Linear Algebra 3rd ed.* Englewood Cliffs, NJ: Prentice-Hall, 1988.

[25] D. G. Luenberger, *Optimization by Vector Space Methods*. New York: Wiley, 1968.

[26] J. A. Sorenson and M. E. Phelps, *Physics in Nuclear Medicine 2nd Ed.* Orlando, FL: Grune and Stratton, 1987.

[27] J. S. Lim, *Two-Dimensional Signal and Image Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1990.

APPENDIX H2

T.-S. Pan and A.E. Yagle, "Numerical Study of Multigrid Implementations of Some Iterative Image Reconstruction Algorithms," IEEE 1991 Medical Imaging Conference, Santa Fe, NM, Nov. 5-9, 1991, pp. 2033-2037.

This is the conference paper version of Appendix H1.

Numerical Study of Multigrid Implementations of Some Iterative Image Reconstruction Algorithms

Tin-Su Pan *and Andrew E. Yagle
Department of Electrical Engineering and Computer Science
The University of Michigan, Ann Arbor, MI 48109-2122

ABSTRACT

The numerical behavior of multigrid implementations of the Landweber, generalized Landweber, ART, and MLEM iterative image reconstruction algorithms is investigated. Comparisons between these algorithms, and with their single-grid implementations, are made on one small-scale synthetic PET system, for phantom objects exhibiting different characteristics. Original contribution is numerical study of the convergence rates of single-grid and multigrid implementations of the Landweber, generalized Landweber, ART, and MLEM iterations.

I. INTRODUCTION

Initialization of an iterative algorithm can be as simple as setting all pixels to zero, or as complex as using the result of filtered back-projection (FBP) [1]. The presumption behind the latter approach is that FBP should furnish a starting point that is "close" to the desirable image, after which the iteration would quickly converge. Surprisingly, this approach has shown little success [1]. In Section III our numerical results lead to a possible explanation of why this is so. We also examine the effects of different initial conditions on the convergence behavior of different iterative algorithms.

The contribution of this paper is a numerical study of *multigrid* implementations of various iterative algorithms: Landweber [2], generalized Landweber [2], ART [3], and MLEM [4]. In a multigrid implementation, the iteration is first used on a coarse grid until it converges; the result is then interpolated and used as an initial condition on a fine grid. The coarse grid iteration requires much less computation per iteration. Although two different system geometry matrices are needed (one for each grid), both matrices are relatively sparse, so the additional storage required for the coarse-grid system matrix is minor.

*The authors wish to thank Dr. W. Leslie Rogers and Mr. Neal Clinthorne of the Division of Nuclear Medicine of the University of Michigan for many helpful discussions and for bringing the generalized Landweber iteration to their attention. The work of the first author was supported in part by NIH grant #PO1-CA42768. The work of the second author was supported in part by ONR grant #N000-14-90-J-1897.

We define a local smoothness property: *the values of four neighboring fine-grid pixels to be grouped as a coarse-grid pixel are close to each other.* Our results indicate that if the image has this property, then the convergence of high-frequency components of the image can be significantly accelerated using a multigrid implementation. "Local smoothness" is not strictly defined here, nor does it need to – the more this property holds, the greater the acceleration of convergence. If the local smoothness property does not hold, or if there are no high-frequency components in the image, then a multigrid implementation does not seem to speed up the convergence rate. Ranganath *et al.* [5] proposed a multigrid implementation of the MLEM algorithm, and gave a numerical example. In this paper, results for several different algorithms are given, more examples are given, and more conclusions about reconstruction behavior are made.

The paper is organized as follows. Section II summarizes the idea of a multigrid implementation and presents the multigrid implementation adopted in this work. Section III presents, summarizes, and discusses numerical results, and presents some conclusions about convergence rates of multigrid implementations. Section IV concludes with a summary.

II. THE MULTIGRID METHOD

The purpose of applying a multigrid method to image reconstruction is to accelerate the iterative reconstruction by first using a coarse-grid iteration to provide an initial condition for the more computationally demanding fine-grid iteration. It was believed [5] that low-frequency components of an image can be reconstructed with less effort using the coarse-grid iteration; then applying the fine-grid iteration results in an efficient reconstruction of high-frequency components.

Ranganath *et al.* [5] used a multigrid method to accelerate the convergence rate of the MLEM algorithm. It was shown that this multigrid-MLEM approach reduced the computation time for reconstructing a 128×128 noisy Shepp-Logan phantom, using a sum of the squared errors criterion. However, the convergence behavior of the multigrid method for image reconstruction in general, was not

considered.

Our multigrid implementation is set up as follows. A coarse-grid is defined, where the center of a coarse-grid pixel is located at the center of the four neighboring fine-grid pixels. Then a coarse-grid system matrix associated with the coarse-grid is computed and stored. Note that the set of projections (elements of b) for the coarse-grid geometry usually does not coincide with the projections of the fine-grid geometry.

In our multigrid method, the projection data are first derived from a noiseless projection of a synthetic image through the fine-grid system. A subset of projection data which corresponds to the set of projections of the coarse-grid system is chosen as the simulated projections from the synthetic image for the coarse grid system. For conservation of total counts, the total counts of the coarse-grid projection data will be normalized such that the total counts of the coarse-grid system will be the same as that of the fine-grid system. Since we are investigating to what extent the multigrid method can contribute to the image reconstruction, we use directly the minimum-norm least-squares solution of the coarse-grid system, which is equivalent to the convergent solution of the coarse-grid iterations (Landweber, generalized Landweber and ART) with zero initial condition. Then the interpolation is applied on the coarse-grid convergent solution to make a fine-grid image. A positivity constraint and another conservation of total counts are applied on the fine-grid image to make up a fine-grid initial image. Finally, the fine-grid iteration with this fine-grid initial condition is executed 1000 times for investigation of its convergence behavior.

III. Numerical Results

A. Experiments

One hypothetical small-scale PET system (PET-1), with a detector ring of radius 1 and a centered object support of 1.2×1.2, is used. The detector ring is equally divided into 26 detectors; and no gap is between any two adjacent detectors. For applying the multigrid method, an object will be pixelated as either $144(= 12 \times 12)$ fine-grid pixels or $36(= 6 \times 6)$ coarse-grid pixels.

A fine-grid system matrix A_1 [6] (dimension 195×144), with 144 pixels, is defined for PET-1, and a coarse-grid system matrix A_1' (dimension 175×36), with 36 pixels, is also defined. SVD analysis shows there are 144 non-zero singular-values for matrix A_1, which indicate that the null space of A_1 is empty.

Fig. 1 shows three images, which are called image-1, image-2 and image-3, separately. Image-1 primarily contains low frequency components, while image-2 and image-3 are explicitly chosen to illustrate the importance of the "local smoothness" property defined before (image-2 does not have this property; image-3 does). Three different initial conditions are investigated: 1) *zero* initial condition, in which every pixel is zero; 2) *average* initial condition, in which every pixel is initialized to be the same value, which is the number of total counts divided by the number of pixels; and 3) ***multigrid*** initial condition, which comes from the interpolation of the minimum-norm least-squares solution of the coarse-grid system. Note that zero initial condition can not be used for the MLEM iteration [4].

B. Results and Discussions

Fig. 2 shows the performance for the Landweber, generalized Landweber, ART, and MLEM iterations with the three images in Fig. 1 and three types of initial conditions (zero, average, multigrid) for PET-1.

The coarse-grid iteration basically estimates the value of a coarse-grid pixel representing its four neighboring fine-grid pixels, which have the same magnitude in image-3, and the interpolation does not violate any edge or boundary of this particular image. Therefore, some high frequency components are picked up from this multigrid procedure for image-3.

We now summarize the results in Fig. 2. For zero or average initial condition (see Fig. 2):

1) The Landweber iteration is usually the slowest of the four methods;

2) The MLEM iteration is usually slower than ART, but faster than the Landweber iteration. However, in some cases, such as image-1 and image-3 (see Fig. 2(a) and 2(c)), the MLEM iterations with average initial condition (in Fig. 2(a)) and with average or multigrid initial condition (in Fig. 2(c)) converge faster than the generalized Landweber and ART iterations.

In order to understand the convergence behavior in Fig. 2, the three images in Fig. 1 are decomposed into components on the singular-vectors v_i of A_1, shown in Fig. 3(a)-(c). There are 144 singular-vectors, ordered by the magnitudes of the singular-values from right to left in the abscissa, with index 144 for the largest singular-value (lowest frequency), and with index 1 for the smallest singular-value (highest frequency). If the image has high frequency components, i.e. non-zero and large components on the left, the iteration tends to take longer to converge.

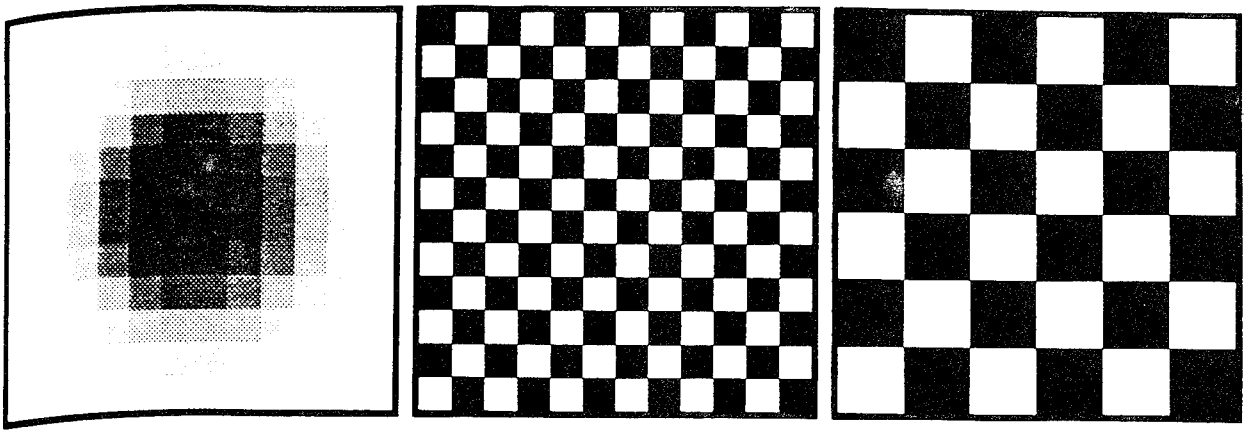
For the multigrid initialization (see Fig. 2):

1) In Fig. 2 (c), the multigrid iteration accelerates the convergence rate dramatically for image-3;

2) In Fig. 2 (a) and (b), the multigrid method does not show any significant improvement over single-grid algorithms for image-1 or image-2.

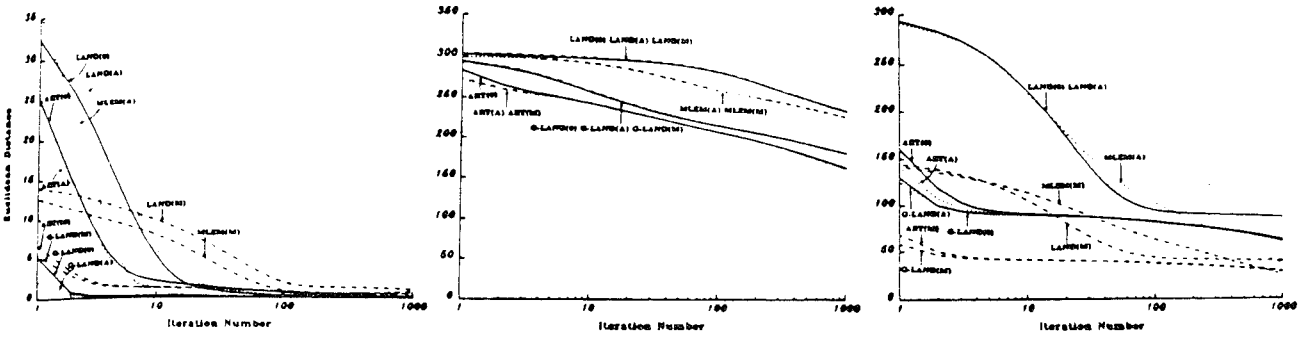
With the aid of the decompositions of the multigrid and average initial conditions for PET-1 in Fig. 3(d)-(i), we can make the following observations:

1) Refer to the decompositions of average initial conditions in Fig. 3(g)-(i). Since the average initial conditions only contain low frequency components (on



(a) (b) (c)

Figure 1: (a) Image-1 (b) image-2, and (c) image-3.



(a) (b) (c)

Figure 2: Euclidean distance between reconstructed and actual images vs. iteration number for (a) image-1, (b) image-2, and (c) image-3 for different algorithms. LAND(0):Landweber iteration with zero initial condition, G-LAND(A):generalized Landweber iteration with average initial condition, etc.

the singular-vectors with biggest singular-values), and these components may be recovered after the first one or two (generalized) Landweber iterations, using the average initial condition may not speed up the convergence for the (generalized) Landweber iteration. For ART, the iteration did not accelerate for these three images for their some pixels are zero or very close to zero. It has been shown [7] that ART may be accelerated by a factor of three by using the average initial condition for the phantom with most of the pixels positive;

2) Compare the decomposition of image-1 (see Fig. 3 (a)) with the decomposition of the multigrid initial condition for image-1 (see Fig. 3 (d)). The multigrid initial condition not only contains some low frequency components, but also introduces some high frequency "noisy" components which will take many iterations to be corrected. In Fig. 2(a), the multigrid Landweber and multigrid MLEM implementations, after several iterations, are slower than their counterparts with zero or average initial condition, and the multigrid generalized Landweber iteration is slower than its counterpart with zero or average initial condition;

3) Compare the decomposition of image-2 (see Fig. 3 (b)) with its corresponding multigrid decomposition (see Fig. 3 (e)). The multigrid initial condition does not recover most of the components, and do not help to accelerate the convergence rate for image-2. The reason for this is as follows. Every coarse-grid pixel is split into 4 equivalent parts to initialize the fine-grid iteration. However, since image-2 does *not* have the local smoothness property defined above, this fine-grid initialization does not resemble the actual image;

4) Compare the decomposition of image-3 (see Fig. 3 (c)) with its corresponding multigrid decomposition (see Fig. 3 (f)). The multigrid initial condition not only contains some low frequency components for image-3, but also contains some high frequency components. Therefore using the multigrid method can greatly accelerate the convergence rate for image-3, which does have the local smoothness property.

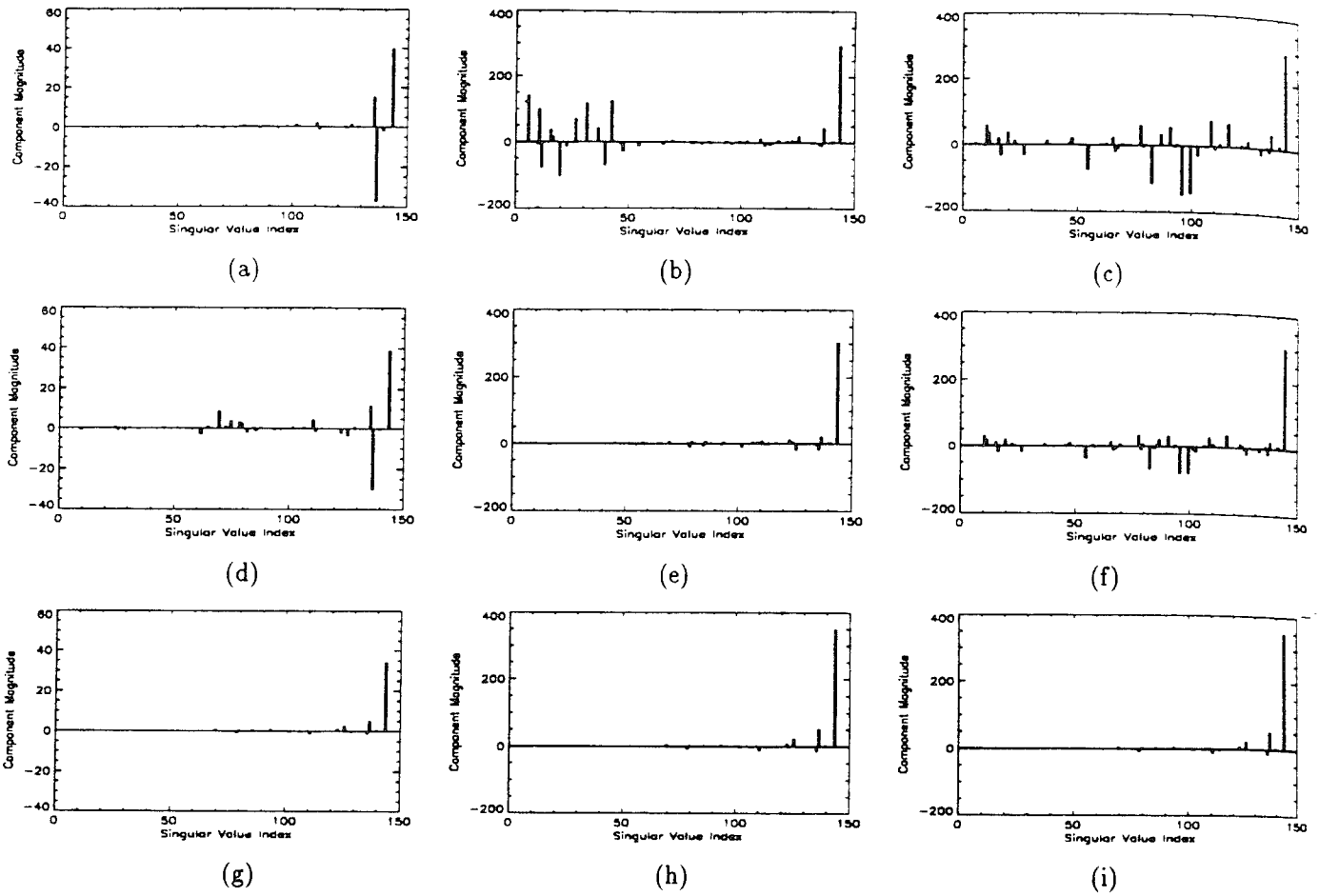


Figure 3: The decompositions of (a) image-1 (b) image-2 and (c) image-3 on the singular-vectors v_i of A_1; the decompositions of the multigrid initial conditions for (d) image-1 (e) image-2 and (f) image-3; and the decompositions of the average initial conditions for (g) image-1 (h) image-2 and (i) image-3.

C. Conclusions

The decompositions in Fig. 3 show that the multigrid method accelerates the convergence rate for high frequency components if the image has the local smoothness property, as image-3 does. The multigrid method does not improve the convergence rate when the image has no high frequency components, as in image-1, or does not have the local smoothness property, as in image-2. Note that *any* image will have the local smoothness property if enough pixels are used to describe it; conversely, the multigrid method could be used to identify images described by too many pixels.

The advantage of using the multigrid method, from our simulation results, is its capability of recovering high frequency components more quickly, if an image has the local smoothness property. It is true that if the coarse-grid iteration recovers some low frequency components of the image for the subsequent fine-grid iteration, then the fine-grid iteration will perform better in the reconstruction of the low frequency components in its first several iterations. However, the fine-grid iteration itself may be very capable of recovering low frequency components, and the benefit of using the coarse-grid iteration to effectively recover low frequencies in the multigrid method may soon disappear after the first several fine-grid iterations. Then the single-grid iteration will be as good as the multigrid iteration.

In some cases, the coarse-grid iteration may be unable to recover high frequency components, and it may even introduce some incorrect high frequency components! Then the multigrid method should show no sign of being able to accelerate the convergence rate after the first several iterations. This may help to explain why the paths of convergence for the multigrid Landweber and multigrid MLEM iterations in Fig. 2 (a) start better than the single-grid implementations in the first several iterations, and then become worse after about the ninth iteration in the Landweber and the fourth in the MLEM.

We refer to this as the "starting good but ending bad" phenomenon, which is due to a non-proper initial condition setting for the fine-grid iteration. This may explain a result in [1], in which using the image from FBP to initialize the conjugate gradient iteration [1] did *not* accelerate the convergence rate of the iteration (see Fig. 4 of [1]). Despite the good starting point, the conjugate gradient

method showed this phenomenon after the 75th iteration. This may be because the FBP initial condition might have introduced some high frequency "noisy" components into the reconstructed image, which dominated the image in the iteration after the low frequency components had been recovered in the first 75 iterations.

D. Comparison with Ranganath's Results

In [5] Ranganath *et al.* proposed a multigrid MLEM algorithm. They claimed that "*the low frequency components of the solution are effectively approximated and recovered on the coarse grids while the high frequency components are recovered on the fine grids*". The first part of the statement about one of the capabilities of the coarse-grid iteration is correct. However, the potential capabilities of the multigrid method to recover high frequency components in the case of local smoothness, and to introduce incorrect high frequency components, were not mentioned. The second part of the statement about the role played by the fine-grid iteration is also correct. However, this role should not be emphasized in using the multigrid method for the following three reasons:

1) For the Landweber and generalized Landweber iterations, the low frequency components recovered by the coarse-grid iteration do not remain unaltered during the fine-grid iteration [6]. Hence the fine-grid iteration must do more than reconstruct high-frequency components;

2) The convergence rate of the fine-grid iteration in recovering high frequency components is not good, unless the local smoothness property also holds (as it might be in the example used in [5]);

3) It is likely that, for images without high frequency components, after the first several iterations the single-grid iteration may be as good as the multigrid iteration, or even better if some incorrect high frequency components were introduced by the coarse-grid iteration.

In interpreting the results of [5] in light of the guidelines proposed in Subsection III.C, the following points should be noted:

1) The original image was represented by 128×128 pixels, which benefits the multigrid method, since the fine sampling induces the local smoothness property;

2) The number of iterations used was small: "four" fine-grid iterations. This again shows the multigrid method in its best light. Since it has received some low frequency components from the coarse-grid iteration, the subsequent fine-grid iteration will seem to have effectively recovered more low frequency components in its first several iterations than the single-grid iteration. The fine-grid iteration may in fact have not done much for the high frequency components in these first several iterations.

The benefit of applying the MLEM multigrid approach on the Shepp-Logan phantom in the first several iterations can also be seen in [6]. Thus the results of [5] seem to be in agreement with the guidelines for the multigrid method.

IV. Summary

A numerical study of multigrid implementations of several iterative image reconstruction algorithms has been presented. The multigrid implementation was found to accelerate the convergence rate of high-frequency components of the image when the image possessed the local smoothness property. In other cases it was unhelpful, and may even slow down the convergence rate. Results of some other papers were interpreted in light of the conclusions drawn here.

References

[1] S. Kawata and O. Nalcioglu, "Constrained iterative reconstruction by the conjugate gradient method," *IEEE Trans. Medical Imaging*, vol. MI-4(2), pp. 65–71, June 1985.

[2] O. N. Strand, "Theory and methods related to the singular-function expansion and Landweber's iteration for integral equations of the first kind," *Siam J. Numer. Anal.*, vol. 11(4), pp. 798–825, Sept. 1974.

[3] Y. Censor, "Finite series-expansion reconstruction methods," *IEEE Proceedings*, vol. 71(3), pp. 409–419, Mar. 1983.

[4] L. A. Shepp and Y. Vardi, "Maximum likelihood reconstruction for emission tomography," *IEEE Trans. Medical Imaging*, vol. MI-1(2), pp. 113–122, Oct. 1982.

[5] M. V. Ranganath, A. P. Dhawan, and N. Mullani, "A multigrid expectation maximization reconstruction algorithm for positron emission tomography," *IEEE Trans. Medical Imaging*, vol. MI-7(4), pp. 273–278, Dec. 1988.

[6] T.-S. Pan and A. E. Yagle, "Numerical study of multigrid implementations of some iterative image reconstruction algorithms," *to appear on IEEE Trans. Med. Imaging*, Dec. 1991.

[7] T.-S. Pan and A. E. Yagle, "Acceleration of Landweber-type algorithms by suppression of projection on the maximum singular vector," *submitted to IEEE Trans. Med. Imaging*, 1991.

APPENDIX I

P. Raadhakrishnan, A.E. Yagle, B.V. Rao, and J.E. Dorband, "On Upper Bounds of the Equivalent Oscillator and Notch-Filter Circuits: A Non-Commutative Group Theoretic Approach," IEEE Trans. Circuits and Systems I 39(9), 756-759, Sept. 1992.

This paper is a minor work that uses group theory to aid in designing oscillator circuits. The application is quite unusual and novel.

Transactions Briefs

On Upper Bounds of the Equivalent Oscillator and Notch-Filter Circuits: A Non-Commutative Group Theoretic Approach

P. Raadhakrishnan, Andrew E. Yagle, B. V. Rao, and John E. Dorband

Abstract—**In this paper we show that the problem of synthesizing equivalent oscillators and notch-filters from the knowledge of a given parent circuit is identical to that of finding the isomorphisms of a non-commutative group. It is also shown that many of the results established by earlier methods can easily be explained using the theory developed in this paper. The results derived in this paper show that earlier results do not lead to the best possible upper bound. The corrected upper bound is derived by counting the members of the non-commutative group. A well-known family of circuits is used to illustrate the theory developed.**

I. Introduction

In the recent past there have been attempts to group *RC* oscillators and notch-filters so that additional equivalent circuits can be generated from the knowledge of a given parent circuit [7]–[14]. Among these efforts, [7]–[12] are applicable only to oscillator circuits. A unified synthesis framework for oscillators and notch-filters alike was presented in [14]. However, the results reported in [15] indicated that the upper bound is greater than the existing bounds.

In this paper a synthesis method based on non-commutative group theory that extends the earlier reported results on upper bounds is presented. Also demonstrated is the fact that the elements of the non-commutative structure are natural extension of the results reported earlier by other investigators. The results reported in this paper have the advantage of synthesizing a stable circuit from an unstable parent circuit.

Let S be a set of n elements $x_1, x_2, x_3, \cdots, x_n$. A 1: 1 mapping Φ acting on S is represented as

$$\Phi: \begin{pmatrix} x_1 & x_2 & x_3 \cdots x_n \\ x_{i1} & x_{i2} & x_{i3} \cdots x_{in} \end{pmatrix} \tag{1}$$

where x_{ik} is the image of x_k under Φ and every $x_{ik} \in S$. A mapping Φ is called decomposable if it can be expressed as a composition of two or more maps. For example, $\Phi = \Phi_2\Phi_1$ means that the action of Φ is equivalent to the action of Φ_1 followed by the action of Φ_2.

II. Theoretical Basis of the Synthesis Problem

2.1. Derivation of the Network Aspects

A popular generalized canonical *RC* oscillator (notch-filter) structure is shown in Fig. 1. The operational amplifier (OA) employed in this circuit is assumed to be ideal with infinite-gain

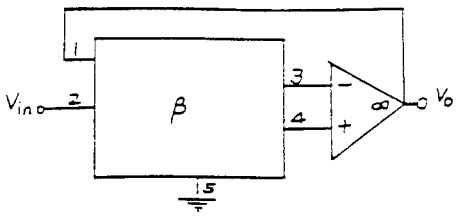


Fig. 1. Generalized canonical *RC* network under study.

mode, and β is assumed to be a passive *RC* network. V_{in} and V_O are the input and the output voltages of the circuit. Since OA is assumed to be ideal with ∞ gain A, ∞ input impedance, and zero output impedance, from Fig. 1 it follows that $A(V_3 - V_4) = V_O$. Since A is infinite and V_O is finite, $V_3 - V_4 = 0$. Hence, the voltages at terminals three and four should be equal. As a consequence, any network connection between terminals three and four is redundant. The condition that the input impedance is ∞ forces the currents $I_3 = I_4 = 0$. Let y_{ij} denote the admittance connected between the nodes i and j of the given passive network β.

1) Oscillator Mode: When the operating mode is that of self-sustained oscillations, $V_{in} = V_O$. With the assumptions made earlier, the following identity results [15]:

$$y_{31} * y_{44} - y_{41} * y_{33} = 0. \tag{2}$$

2) Notch-Filter Mode: For the notch-filter mode, at transmission null frequency ω_O, the output voltage V_O is zero for a nonzero finite input voltage V_{in}. This condition, along with the additional observations made earlier, results in

$$y_{32} * y_{44} - y_{42} * y_{33} = 0. \tag{3}$$

Let λ be a set defined as λ: $\{\theta_1, \theta_2, \gamma_1, \gamma_2\}$ where the elements of λ satisfy the constraint

$$\theta_1 * \theta_2 - \gamma_1 * \gamma_2 = 0. \tag{4}$$

Set λ can be used to denote the equations (2) or (3) due to their structural similarity. Depending on whether the circuit is an oscillator or a notch-filter, λ will take values from set $\{y_{31}, y_{44}, y_{41}, y_{33}\}$ or set $\{y_{32}, y_{44}, y_{42}, y_{33}\}$, respectively. This observation leads to a unified approach to the synthesis problem as described in the next section.

III. Relationship Between Group Isomorphisms of λ and Synthesis of Equivalent Circuits

Consider set λ: $\{\theta_1, \theta_2, \gamma_1, \gamma_2\}$ where the elements of λ satisfy constraint (4), which is called the reduced identity in the context of circuit theory [14]. Since this identity is common to all the

Manuscript received August 12, 1991; revised June 8, 1992. This work was supported by the Office of Naval Research under Grant N00014-90-J-1897. This paper was recommended by Associate Editor D. J. Allstot.

The authors are with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109.

IEEE Log Number 9204227.

1057-7122/92$03.00 © 1992 IEEE

oscillators (notch-filters) that can be represented by the two-port structure in Fig. 1, any isomorphism of λ that leaves (4) unchanged will lead to an additional oscillator (notch-filter) circuit. Hence, set S of all isomorphisms of λ that leaves constraint (4) unchanged will enable one to find the upper bounds on the additional non-trivial equivalent circuits. In order to illustrate this point, let Φ_i be an element of set S. Action of Φ_i on λ will lead to $\Phi_i(\lambda)$: $\{\Phi_i(\theta_1), \Phi_i(\theta_2), \Phi_i(\gamma_1), \Phi_i(\gamma_2)\}$. Constraint (4) is changed to the form

$$\Phi_i(\theta_1) * \Phi_i(\theta_2) - \Phi_i(\gamma_1) * \Phi_i(\gamma_2) = 0. \tag{5}$$

Since λ has only four elements, the following are all the possible invariant isomorphisms:

$$\Phi_1: \begin{pmatrix} \theta_1 & \theta_2 & \gamma_1 & \gamma_2 \\ \theta_1 & \theta_2 & \gamma_1 & \gamma_2 \end{pmatrix}; \Phi_2: \begin{pmatrix} \theta_1 & \theta_2 \\ \theta_2 & \theta_1 \end{pmatrix}; \Phi_3: \begin{pmatrix} \gamma_1 & \gamma_2 \\ \gamma_2 & \gamma_1 \end{pmatrix};$$

$$\Phi_4: \begin{pmatrix} \theta_1 & \theta_2 & \gamma_1 & \gamma_2 \\ \theta_2 & \theta_1 & \gamma_2 & \gamma_1 \end{pmatrix};$$

$$\Phi_5: \begin{pmatrix} \theta_1 & \theta_2 & \gamma_1 & \gamma_2 \\ \gamma_1 & \gamma_2 & \theta_1 & \theta_2 \end{pmatrix};$$

$$\Phi_6: \begin{pmatrix} \theta_1 & \theta_2 & \gamma_1 & \gamma_2 \\ \gamma_2 & \gamma_1 & \theta_2 & \theta_1 \end{pmatrix}; \Phi_7: \begin{pmatrix} \theta_1 & \theta_2 & \gamma_1 & \gamma_2 \\ \gamma_1 & \gamma_2 & \theta_2 & \theta_1 \end{pmatrix};$$

$$\Phi_8: \begin{pmatrix} \theta_1 & \theta_2 & \gamma_1 & \gamma_2 \\ \gamma_2 & \gamma_1 & \theta_1 & \theta_2 \end{pmatrix}.$$

The following interesting properties are observed from the eight isomorphisms derived:

$$\Phi_1 = e = \text{identity mapping.}$$

$$\Phi_i * \Phi_i = \Phi_i^2 = e,\ i \in \{1, 2, 3, \cdots, 8\}$$

$$\Phi_2 * \Phi_3 = \Phi_3 * \Phi_2 = \Phi_4; \quad \Phi_5 * \Phi_6 = \Phi_6 * \Phi_5 = \Phi_4; \quad \Phi_2 * \Phi_3 = \Phi_3 * \Phi_2 = \Phi_4;$$

$$\Phi_3 * \Phi_4 = \Phi_4 * \Phi_3 = \Phi_2; \quad \Phi_6 * \Phi_4 = \Phi_4 * \Phi_6 = \Phi_5; \quad \Phi_3 * \Phi_4 = \Phi_4 * \Phi_3 = \Phi_2;$$

$$\Phi_4 * \Phi_2 = \Phi_2 * \Phi_4 = \Phi_3; \quad \Phi_4 * \Phi_5 = \Phi_5 * \Phi_4 = \Phi_6; \quad \Phi_4 * \Phi_2 = \Phi_2 * \Phi_4 = \Phi_3.$$

It is easily verified that sets S_1, S_2, and S_3 defined as S_1: $\{\Phi_1, \Phi_2, \Phi_3, \Phi_4\}$, S_2: $\{\Phi_1, \Phi_4, \Phi_5, \Phi_6\}$, and S_3: $\{\Phi_1, \Phi_4, \Phi_7, \Phi_8\}$ satisfy all the conditions of the Klein group presented in [14].

Another interesting property is that all the isomorphisms are self inverses. Hence, repeated application of any isomorphism on λ will be equivalent to applying the identity mapping or the original mapping itself, depending on whether the number of times the mapping is applied is odd or even, respectively. From a circuit theory viewpoint, this implies that the repeated application of a particular admittance transformation will not lead to more than one additional equivalent circuit. The same conclusion can be made by considering the two-element group, i.e., the identity element and any other given isomorphism from set S.

In order to construct the necessary non-commutative group, we have to note the following additional properties exhibited by the elements of set S:

$$\Phi_3 * \Phi_6 = \Phi_6 * \Phi_2 = \Phi_2 * \Phi_5 = \Phi_5 * \Phi_3 = \Phi_8$$

$$\Phi_3 * \Phi_5 = \Phi_5 * \Phi_2 = \Phi_2 * \Phi_6 = \Phi_6 * \Phi_3 = \Phi_7$$

$$\Phi_5 * \Phi_7 = \Phi_7 * \Phi_6 = \Phi_6 * \Phi_8 = \Phi_8 * \Phi_5 = \Phi_2$$

$$\Phi_7 * \Phi_5 = \Phi_6 * \Phi_7 = \Phi_8 * \Phi_6 = \Phi_5 * \Phi_8 = \Phi_3$$

$$\Phi_7 * \Phi_2 = \Phi_3 * \Phi_7 = \Phi_2 * \Phi_8 = \Phi_8 * \Phi_3 = \Phi_5$$

$$\Phi_2 * \Phi_7 = \Phi_7 * \Phi_3 = \Phi_8 * \Phi_2 = \Phi_3 * \Phi_8 = \Phi_6.$$

Since the elements of S are self inverses, any additional combination of the maps can be reduced to one of the above given forms. Hence, S is closed under the composition operation. A careful analysis shows that a composition of any two elements from different subsets S_1, S_2, and S_3 is noncommutative. The composition operation plays the role of the binary operation for the group S. A number of nontrivial new solutions to condition (4) can now be computed using the group theoretic approach in more than one way. Given that the elements of S are the restricted isomorphisms of λ, the number of nontrivial additional solutions is equal to [{cardinality of S} − 1] (the [−1] being the removal of the identity element from S). Hence, the number of additional new solutions is equal to seven.

The elements of λ take values from the two-port admittances that satisfy (2) for the oscillator mode (or (3) for the notch-filter). Therefore, computing all possible invariant isomorphisms of λ is equivalent to finding all the possible different two-port admittance transformations that leave the condition for oscillations unaffected. Hence, the number of additional equivalent circuits is equal to [{cardinality of S} − 1].

IV. Circuit Theoretic Interpretation of the Properties of the Non-Commutative Group

1) Φ_1 is an Identity Mapping: After having chosen set λ: $\{\theta_1, \theta_2, \gamma_1, \gamma_2\}$ from the sets $\{y_{31}, y_{44}, y_{41}, y_{33}\}$ or $\{y_{32}, y_{44}, y_{42}, y_{33}\}$, applying Φ_1 to λ leads to the set $\Phi_1(\lambda)$: $\{\Phi_1(\theta_1), \Phi_1(\theta_2), \Phi_1(\gamma_1), \Phi_1(\gamma_2)\}$. However, since Φ_1 is the identity mapping ($\lambda = \Phi_1(\lambda)$), condition (4) is unaffected by the action of Φ_1. This means that when there is no rearrangement of the parent circuit, the original circuit will remain unaffected and there will be no new equivalent circuit.

2) Every Element of Set S is a Self Inverse: When $i \neq 1$, Φ_i, for $i = 2, 3, \cdots, 8$ will be a nontrivial isomorphism from $(S, *)$, where $*$ is the composition operation. The action of Φ_i on λ leads to reordering of the elements of λ and leave condition (4) fixed. Since λ contains the two-port admittances of the parent circuit, any reordering of λ leads to the reordering of the admittances and results in an additional equivalent circuit. This means that given a parent oscillator circuit, the action of a nontrivial Φ_i readily synthesizes an additional new oscillator (notch-filter) circuit with the same operating frequency. However, if Φ_i is applied to the new set $\Phi_i(\lambda)$, the resulting set is $\Phi_i * (\Phi_i(\lambda))$ which can be rewritten as $\Phi_i^2(\lambda) = \Phi_1(\lambda) = \lambda$. Clearly the application of Φ_i again will result in the set $\Phi_i(\lambda)$. This leads to the conclusion that a given nontrivial isomorphism leads to a unique equivalent circuit. Hence, the seven nontrivial isomorphisms lead to seven nontrivial equivalent circuits. This is not sufficient to conclude that the upper bound on the additional equivalent oscillator or notch-filter circuits is seven. The next observation leads to the sufficient part of the upper bound.

3) Set S Forms a Non-Commutative Finite Group: It was noted that the elements of set S form a non-commutative group. Set S can be written as the union of subsets S_1, S_2 and S_3 which form three different Klein groups. Hence, the total number of elements of the group S can be easily computed using the set

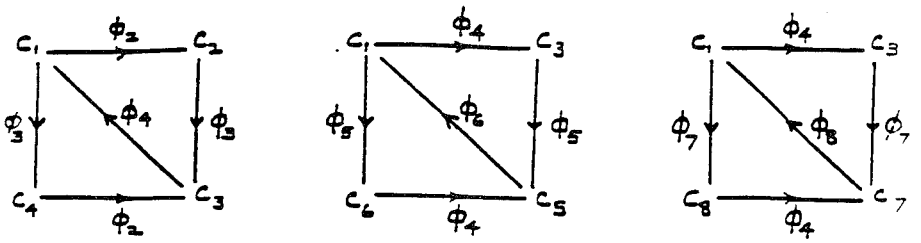


Fig. 2. Group isomorphic diagrams for oscillators.

theoretic concepts as

$$|S_1| + |S_2| + |S_3| - |S_1 \cup S_2| - |S_2 \cup S_3|$$

$$- |S_3 \cup S_1| + |S_1 \cap S_2 \cap S_3|$$

$$= 4 + 4 + 4 - 2 - 2 - 2 + 1 = 7.$$

This is much easier than going though an exhaustive search of all the possible combinations of the elements of S. It can be verified that there are 120 different ways in which these elements can be combined. Since some of them are non-commutative mappings, it is possible to have infinitely many different compositions which are equivalent to one of the eight elements of S. Hence, the number of nontrivial equivalent circuits is seven.

4) Subset S_1 is Complement to the Set $S_2 \cup S_3$ in the Sense of Stability: It can be shown that if the parent circuit is a stable circuit, subgroup S_1 will generate all stable circuits and subgroups S_2 and S_3 will generate some unstable circuits. Since S has eight elements and the subgroup S_1 has four elements, the number of stable circuits from a stable parent circuit will be four. This result holds for an oscillator and a notch-filter alike. A restricted version of this result for the case of a stable parent oscillator is available in [7]. Hence, even for a stable parent circuit, the results derived here are more general. However, the results gain additional strength from the proof that the present method can produce the identical number of stable circuits (notch-filters and oscillators) starting from an unstable parent circuit. This result is new and provides a tight upper bound on the stable circuits that can be synthesized from a stable or an unstable parent circuit. In the restricted case of oscillators, given an unstable parent circuit, results in [7] will produce only unstable circuits. Hence the result in [7] needs an additional condition that the parent circuit should be stable. Stability is not an issue in our procedure, as explained above. The next subsection shows how the non-commutative group results generalize the earlier reported results.

V. Interpretation of the Earlier Results From Non-Commutative Framework

Most of the earlier results are applicable to either an oscillator or a notch-filter. Since the non-commutative result is applicable to both cases it is of interest to find out how well the established synthesis techniques relate to the present theory. Decomposition of set S into the commutative subgroups S_1, S_2, and S_3 becomes useful in this discussion.

In [7] it was shown that given a parent circuit employing an ideal OA, it is possible to generate three additional equivalent oscillators with the same operating frequency. The key assumption not explicitly mentioned in [7] is that the parent oscillator circuit needs to be stable. Hence, this method forces the inventor to find at least one stable circuit by trial and error method. Given an unstable circuit, the method in [7] will generate only unstable circuits because it preserves the transfer function form. It can be shown that the method in [7] is identical to using subgroup S_1 for a given stable parent circuit. Since the details will unduly extend the length of the paper we omit the proofs; they will be presented elsewhere.

Results in [8] can be obtained by choosing only S_1 and S_2 to construct additional equivalent circuits from the knowledge of a parent circuit. The upper bound on the additional equivalent oscillators was shown to be five in [8], and the nontrivial isomorphisms for sets S_1 and S_2 taken together were shown to be five in our earlier paper [14]. Since the results presented in [14] form a subset of the results in this paper, we conclude that taking any two subgroups from S_1, S_2, and S_3 leads to the upper bounds derived in [8]. The group isomorphic diagrams are presented in Fig. 2.

VI. Illustrative Example

In this section we illustrate the theory developed in this paper and the claim that the non-commutative group method will enable one to synthesize the stable as well as unstable circuits for a given parent circuit. Since there are well-established results available for the case of oscillators we illustrate the theory for oscillators.

Assume that we are given the circuit in Fig. 3(a). In terms of the two-port admittances, the necessary constraint equation can be derived as

$$y_1 * y_3 - y_2 * y_4 = 0.$$

At this stage there are two choices for the admittances, as given below:

1)

$$y_4 = \frac{SC_4}{SC_4R_4 + 1}, y_2 = \frac{1}{R_2}, y_3 = R_3 \text{ and } y_1 = \frac{1}{R_1} + SC_1$$

2)

$$y_1 = \frac{SC_1}{SC_1R_1 + 1}, y_2 = \frac{1}{R_2}, y_3 = R_3 \text{ and } y_4 = \frac{1}{R_4} + SC_4.$$

Though both of these choices lead to the necessary conditions, only one choice leads to a stable circuit. In general, stability cannot be predetermined. Circuits have to be generated and then using simulation methods, the stable circuits are chosen. Let us consider the first choice of parameters. The corresponding circuit is shown in Fig. 3(a). Letting $\theta_1 = y_1$, $\theta_2 = y_3$, $\gamma_1 = y_2$, and $\gamma_2 = y_4$, we can construct the necessary non-commutative

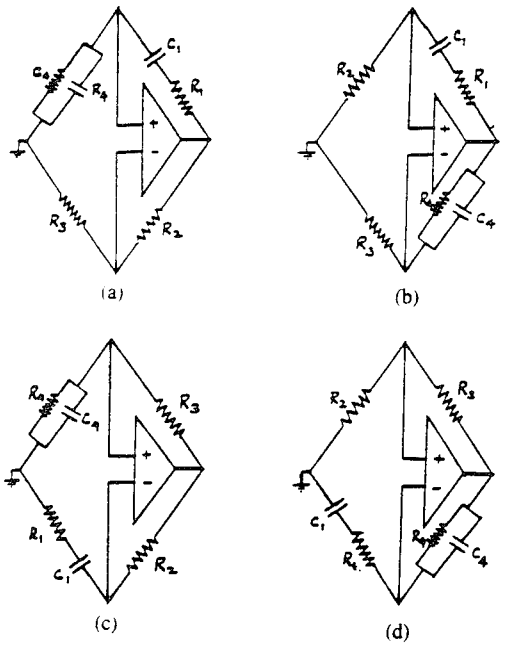


Fig. 3. Unstable Wien-Bridge oscillator family.

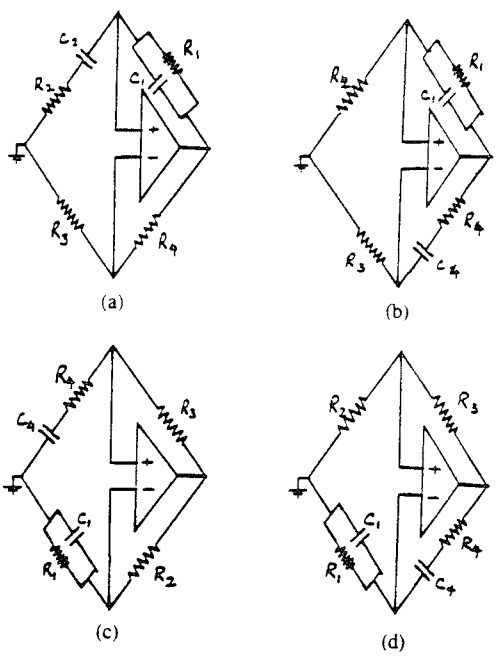


Fig. 4. Stable Wien-Bridge oscillator family.

group S. Elements of S are defined as follows:

$$\Phi_1: \begin{pmatrix} \theta_1 & \theta_2 & \gamma_1 & \gamma_2 \\ \theta_1 & \theta_2 & \gamma_1 & \gamma_2 \end{pmatrix}; \Phi_2: \begin{pmatrix} \theta_1 & \theta_2 \\ \theta_2 & \theta_2 \end{pmatrix}$$

$$\Phi_3: \begin{pmatrix} \gamma_1 & \gamma_2 \\ \gamma_2 & \gamma_1 \end{pmatrix}; \Phi_4: \begin{pmatrix} \theta_1 & \theta_2 & \gamma_1 & \gamma_2 \\ \theta_2 & \theta_1 & \gamma_2 & \gamma_1 \end{pmatrix}$$

$$\Phi_5: \begin{pmatrix} \theta_1 & \theta_2 & \gamma_1 & \gamma_2 \\ \gamma_1 & \gamma_2 & \theta_1 & \theta_2 \end{pmatrix}; \Phi_6: \begin{pmatrix} \theta_1 & \theta_2 & \gamma_1 & \gamma_2 \\ \gamma_2 & \gamma_1 & \theta_2 & \theta_1 \end{pmatrix}$$

$$\Phi_7: \begin{pmatrix} \theta_1 & \theta_2 & \gamma_1 & \gamma_2 \\ \gamma_1 & \gamma_2 & \theta_2 & \theta_1 \end{pmatrix}; \Phi_8: \begin{pmatrix} \theta_1 & \theta_2 & \gamma_1 & \gamma_2 \\ \gamma_2 & \gamma_1 & \theta_1 & \theta_2 \end{pmatrix}.$$

Furthermore, sets S_1, S_2, and S_3 defined as S_1: $\{\Phi_1, \Phi_2, \Phi_3, \Phi_4\}$, S_2: $\{\Phi_1, \Phi_4, \Phi_5, \Phi_6\}$, and S_3: $\{\Phi_1, \Phi_4, \Phi_7, \Phi_8\}$ satisfy all the conditions of a Klein group [14].

Action of Φ_2 on the circuit shown in Fig. 3(a) leads to the circuit shown in Fig. 3(b). Action of Φ_3 on the circuits in Figs. 3(a) and (b) leads to the circuits shown in Figs. 3(c) and (d), respectively. It can be verified that the elements of subgroups S_1 do not generate any additional circuits. We now show that the results in [7] will generate identical circuits. Applying *theorem* 1 in [7] to the circuit in Fig. 3(a), we can generate the circuit in Fig. 3(b). Applying *theorem* 2 in [7] to circuits in Figs. 3(a) and (b) leads to the circuits in Figs. 3(c) and (d), respectively. Since the method in [7] preserves the transfer function, it cannot generate any additional circuits.

However, the non-commutative group theoretic method can generate additional circuits using subgroups S_2 and S_3. Action of Φ_5, Φ_6, Φ_7, and Φ_8 on the circuit in Fig. 3(a) generates the circuits in Figs. 4(a), (b), (c), and (d), respectively. The circuits in Fig. 4 are all stable members of the well-known Wien Bridge oscillator family. We also note that applying subgroup S_1 to the circuit in Fig. 4(a) leads to the remaining circuits in Fig. 4. Hence, applying the results in [7] to the circuit in Fig. 4(a) will lead to the remaining circuits in Fig. 4.

A vast number of oscillator circuits are reported in [8]. These circuits can be generated and grouped using the theory presented here.

REFERENCES

[1] S. K. Mitra, *Active Inductorless Filters*. New York: IEEE Press, 1971.
[2] R. Schaumann, M. Soderstrand, and K. Laker, *Modern Active Filter Design*. New York: IEEE Press, 1981.
[3] S. K. Mitra, *Analysis and Synthesis of Linear Active Networks*. New York: Wiley, 1969.
[4] G. S. Moschytz, *Linear Integrated Networks*, vol. 2. New York: Van Nostrand Reinhold, 1974.
[5] L. T. Brouton, *RC Active Circuits Theory and Design*. Englewood Cliffs, NJ: Prentice-Hall, 1980.
[6] W. K. Chen, "Analysis of constraint active networks," *Proc. IEEE*, vol. 66, pp. 1655–1657, 1978.
[7] N. Boutin, "On the identification of single amplifier single resistance controlled oscillators," *IEEE Trans. Circuits Syst.*, vol. CAS-31, pp. 1046–1048, Dec. 1984.
[8] B. B. Bhattacharya, M. Sundara Murthy, and M. N. S. Swamy, "Systematic generation of canonical sinusoidal RC-active oscillators," *Proc. Inst. Elec. Eng.*, vol. 128, pp. 114–125, June 1981.
[9] D. Hilberman, "Input and ground as components in active filters," *IEEE Trans. Circuit Theory*, vol. CT-20 pp. 540–547, Sept. 1973.
[10] D. Hilberman, "An approach to the sensitivity and statistical variability of biquadratic filters," *IEEE Trans. Circuit Theory*, vol. CT-20, pp. 382–390.
[11] Wierzba, "Op-amp relocation: A topological active network synthesis," *IEEE Trans. Circuits Syst.*, pp. 469–475, 1986.
[12] T. S. Rathore, B. B. Bhattacharya, "Derivation of sinusoidal oscillators through network transformations," *JIETE*, vol-34, pp. 455–457, Dec. 1988.
[13] P. Raadhakrishnan, *Systematic Realization of a Pair of Imaginary Axis Transmission Zeros, Internal Seminar Report*, Bombay: Indian Institute of Technology, 1987.
[14] P. Raadhakrishnan and B. V. Rao, "Upper bound on the number of equivalent oscillator-notch filter circuits: A group theoretic approach," *Int. J. Circuit Theory Appl.*, vol. 18, pp. 521–533, 1990.
[15] V. V. Prem Pyara, S. C. Dutta Roy, and S. S. Jamuar, "Identification and design of single amplifier single resistance controlled oscillators," *IEEE Trans. Circuits Syst.*, vol. CAS-30, pp. 176–181, 1983.

APPENDIX J1

B. Sahiner and A.E. Yagle, "Image Reconstruction from Projections Under Wavelet Constraints," to appear in IEEE Trans. Sig. Proc. 41(12), December 1993 (special issue on wavelets).

This paper considers the problem of image reconstruction from projections, given constraints not on the image, but on certain wavelet coefficients of the image. The idea is that low-resolution regions of the image can be locally low-pass filtered by setting high-resolution wavelet coefficients to zero. These are then used as constraints on the image reconstruction process, so that other areas of the reconstructed image are improved as well. The constraints are implemented as a simple filter directly on the image.

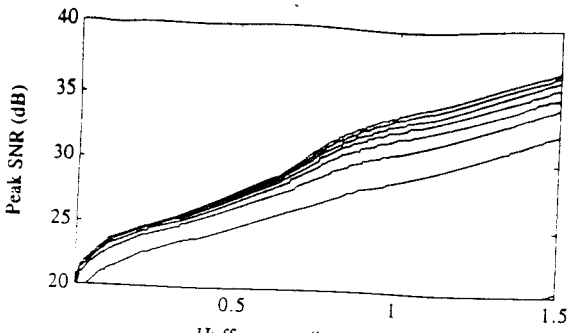


Fig. 10. Peak SNR versus Huffmann bit rate, for different lengths $L = 4, 6, \cdots, 18$ and $K = L/2 - 2$. The initial image was BARBARA.

References

[1] M. Antonini, M. Barlaud, and P. Mathieu, "Image coding using lattice vector quantization of wavelet coefficients," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Toronto, Canada, 1991, pp. 2273–2276.

[2] P. J. Burt and E. H. Adelson, "The Laplacian pyramid as a compact image code," *IEEE Trans. Commun.*, vol. 31, pp. 532–540, Apr. 1983.

[3] R. E. Crochiere, S. A. Weber, and J. L. Flanagan, "Digital coding of speech in subbands," *Bell Syst. Tech. J.*, vol. 55, pp. 1069–1085, 1991.

[4] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Commun. Pure Appl. Math.*, vol. XLI, no. 7, pp. 909–996, 1988.

[5] ——, "Orthonormal bases of compactly supported wavelets II. Variations on a theme," *SIAM J. Math. Anal.*, 1993, to appear.

[6] I. Daubechies and J. C. Lagarias, "Two-scale difference equations II.," *SIAM J. Math. Anal.*, vol. 23, pp. 1031–1079, July 1992.

[7] C. Dorize and L. F. Villemoes, "Optimizing time-frequency resolution of orthonormal wavelets," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Toronto, Ontario, May 1991, pp. 2029–2032.

[8] N. Dyn, "Subdivision schemes in CADG," in W. A. Light, Ed., *Advances in Numerical Analysis. II. Wavelets, Subdivision Algorithms and Radial Functions.* London: Oxford, 1991, pp. 36–104.

[9] O. Hermann, "On the approximation theorem in nonrecursive digital filter design," *IEEE Trans. Circuit Theory*, vol. 18, pp. 411–413, May 1971.

[10] T. Kronander, Some aspects of perception based image coding, Ph.D. dissertation, Linköping Univ., Sweden, 1989.

[11] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 11, pp. 674–693, July 1989.

[12] K. Ramchandran and M. Vetterli, "Best wavelet packet bases in a rate-distortion sense," *IEEE Trans. Image Processing*, vol. 2, pp. 160–175, Apr. 1992.

[13] O. Rioul, "A simple, optimal regularity estimate for wavelets," in *Proc. EUSIPCO*, Brussels, Belgium, vol. II, Sept. 1992, pp. 937–940.

[14] ——, "Simple regularity criteria for subdivision schemes," *SIAM J. Math. Anal.*, vol. 23, pp. 1544–1576, Nov. 1992.

[15] O. Rioul and P. Duhamel, "A Remez exchange algorithm for orthonormal wavelets," *IEEE Trans. Circuits Syst.*, 1993, submitted.

[16] O. Rioul and M. Vetterli, "Wavelets and signal processing," *IEEE Signal Processing Mag.*, vol. 8, Oct. 1991.

[17] M. J. T. Smith and T. P. Barnwell, "Exact reconstruction techniques for tree-structured subband coders," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 434–441, June 1986.

[18] M. Vetterli, J. Kovačević, and D. L. Gall, "Perfect reconstruction filter banks for HDTV representation and coding," *Image Commun.*, vol. 2, pp. 349–364, Oct. 1990.

[19] H. Volkmer, "On the regularity of wavelets," *IEEE Trans. Informat. Theory*, vol. 38, pp. 872–876, Mar. 1992.

Image Reconstruction from Projections under Wavelet Constraints

Berkman Sahiner and Andrew E. Yagle

Abstract—First, we discuss how the wavelet transform can be used to perform spatially-varying filtering of an image, suppressing noise locally in smooth regions of the image, and we discuss detection of such regions in a noise-corrupted image. Second, we show how to compute the minimum mean-square estimate of an image given: 1) noisy projections of the image; 2) statistics of additive noise in the projections; and 3) constraints on wavelet coefficients of the image. Examples illustrate the resulting procedure.

I. Introduction

In many problems arising in fields such as medical imaging, nondestructive testing, radio astronomy, and geophysics [1], one needs to reconstruct a two-dimensional object or image from its projections, which amounts to computing the inverse Radon transform. A problem with the inverse Radon transform is that the ramp filter amplifies the high-frequency components of both the noise and the data. Since noise usually dominates at high frequencies, it is common practice to use a low-pass filter in conjunction with the ramp filter to improve the signal-to-noise ratio (SNR). However, the SNR improvement obtained by using a low-pass filter comes at the expense of degraded image resolution, since high-resolution features in the image will also be smoothed. It is desirable to reduce the noise energy in the reconstructed image over regions where high-resolution features are not present, by using spatially-varying filtering.

In this note, we use wavelets to perform this desired localized low-pass filtering. We show how thresholding can be used to determine the regions in the wavelet domain where wavelet coefficients may be set to zero, effecting spatially-varying filtering, and provide a statistical justification for it. Alternatively, *a priori* information about the image can be used to identify such regions. We then use these zero wavelet coefficients as constraints, and compute the minimum mean-squared error image which satisfies these constraints.

II. The Radon and Wavelet Transforms

A. Image Reconstruction From Projections

The inverse Radon transform problem is to reconstruct an image $\mu(x, y)$ from its projections $p(r, \theta)$ where

$$p(r, \theta) = \mathcal{R}\{\mu(x, y)\} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \mu(x, y)\, \delta(r - x \cos\theta - y \sin\theta)\, dx\, dy \quad (1)$$

is the Radon transform of $\mu(x, y)$. A common procedure for obtaining $\mu(x, y)$ from $p(r, \theta)$ is filtered backprojection (FBP), in which the projections are first filtered to yield $s(r, \theta) = p(r, \theta) *$

Manuscript received August 28, 1992; revised May 19, 1993. The Guest Editor coordinating the review of this paper and approving it for publication was Prof. Martin Vetterli. This work was supported in part by the Office of Naval Research under Grant N00014-90-J-1897.

The authors are with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109-2122.

IEEE Log Number 9212174.

1053-587X/93$03.00 © 1993 IEEE

$q(r)$, where the Fourier transform of the ramp filter $q(r)$ approximates $|w|$. The image $\mu(x, y)$ is then obtained by backprojecting $s(r, \theta)$. In practical problems, we have only samples of $p(r, \theta)$, in which case samples of the image $\mu(i, j)$ are obtained by a discretization of the continuous FBP method

$$\mu(i, j) = \mathcal{R}_d^{-1}\{p(m, n)\} = \frac{1}{2N}\sum_{n=0}^{N-1}\sum_{m=-\infty}^{\infty}\sum_{l=-\infty}^{\infty} q(l-m) \cdot p(m, n)h(i\cos n\Delta + j\sin n\Delta - l) \quad (2)$$

where N is the total number of views, $\Delta = \pi/N$, $p(m, n)$ is the mth projection in the nth view, $h(x)$ is an interpolation function (e.g., for linear interpolation $h(x) = 1 - |x|$ for $|x| < 1$ and $h(x) = 0$ for $|x| \geq 1$), and $\mathcal{R}_d^{-1}$ is the discrete version of the inverse Radon transform operator.

B. The Wavelet Transform

The wavelet transform can be viewed as a time-frequency representation that has good localizing properties. In this correspondence, we restrict attention to orthogonal, separable 2-D wavelet transforms [2]. We assume that the reader is familiar with the wavelet transform theory and we only establish the notation to be used.

The 2-D wavelet transform $W_{2^j}^i f(x, y)$ of a 2-D, square-integrable function $f(x, y)$ is defined as

$$W_{2^j}^i f(n, m) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(z_1, z_2) \cdot \psi_{2^j}^i(2^j n - z_1, 2^j m - z_2)\, dz_1\, dz_2 \quad (3)$$

where $\psi_{2^j}^i(x, y) = 2^{-j}\psi^i(2^{-j}x, 2^{-j}y)$, and $\psi^i(x, y)$, $1 \leq i \leq 3$ are the orthonormal subwavelets with different orientations in the 2-D plane for different i.

For a discrete 2-D function $f_o(n, m)$, the wavelet transform is defined as follows. First, the blurred signal at scale j is defined recursively as

$$f_j(n, m) = \sum_{k_1}\sum_{k_2} h(2n - k_1)h(2m - k_2)f_{j-1}(k_1, k_2) \quad (4)$$

and then the wavelet transform at scale j is defined as

$$W_{2^j}^i f(n, m) = \sum_{k_1}\sum_{k_2} g^i(2n - k_1, 2m - k_2)f_{j-1}(k_1, k_2), \quad i = 1, 2, 3 \quad (5)$$

where $g^1(n, m) = g(n)h(m)$, $g^2(n, m) = g(m)h(n)$, and $g^3(n, m) = g(n)g(m)$. For a detailed description of properties satisfied by the wavelets $\psi_{2^j}^i(x, y)$ and the filters $g(n)$ and $h(n)$, see [2], [3].

III. Multiscale Filtering

The idea of filtering a signal in time–frequency space has been discussed and employed for the Wigner distribution and short-time Fourier transform in [4]–[6]. In this note, we regard the wavelet transform as a time frequency representation and perform a simple version of multiscale filtering by setting the wavelet coefficients of the signal to zero in regions of the wavelet transform domain where the signal energy is known or estimated to be much smaller than the noise energy. Since fine-scale wavelet transform components represent localized high-resolution features of the image, windowing these to zero effectively smoothes the image, much as low-pass filtering does. The advantage of using wavelets is that this can be done on a localized basis, smoothing in some areas while leaving other areas (such as edges) unaffected.

In image reconstruction, there is often considerable *a priori* knowledge about the image, and the region in the wavelet domain where we set the wavelet coefficients to zero can be determined from this knowledge. For example, it may be known, *a priori*, that there are no high-resolution features in some region D of the image. D may be known to represent a flat or slowly-varying part of the image, or D may be known to be free of edges. Since fine-scale wavelet transform components represent localized high-resolution features of the image, we can window these to zero in the region D, and cancel some of the noise energy without degrading the image.

If there is no *a priori* knowledge of image features, a thresholding approach may be used to set some wavelet coefficients to zero. The idea is to eliminate noise where it is possible to do so without significantly degrading the image. Since the wavelet transform is being used, this noise filtering can be done on a localized basis.

The ideas of using thresholding on a time-frequency representation and setting certain regions of a time-frequency representation to zero for time-varying filtering have been mentioned and used in [5]–[9]. Here, we supply a statistical justification for the thresholding approach as we use it in this correspondence.

A. Detection Problem Formulation

Assume that $f_o(x, y)$ is a zero-mean white Gaussian random field with power spectral density $\sigma_{f_o}^2$, and let $\{\psi_{2^j}^i(2^j n - x, 2^j m - y), i = 1, 2, 3, (j, n, m) \in Z^3\}$ be an orthonormal wavelet family. Let $W_{2^j}^i f_o(n, m)$ be the discrete wavelet transform of $f_o(x, y)$, defined using (3). Then, by the orthonormality of the wavelet family, the quadruply indexed random sequence $W_{2^j}^i f_o(n, m)$ is uncorrelated and zero-mean Gaussian, with variance $\sigma_{f_o}^2$. To obtain a random field $f(x, y) \neq f_o(x, y)$ whose wavelet coefficients are zero with probability one outside a region D_1, we define

$$f(x, y) = \sum_{j,n,m,i \in D_1} W_{2^j}^i f_o(n, m)\psi_{2^j}^i(2^j n - x, 2^j m - y). \quad (6)$$

We now state the problem. Given the noisy observations

$$f_\eta(x, y) = f(x, y) + \eta(x, y) \quad (7)$$

of $f(x, y)$, where $\eta(x, y)$ is a zero-mean white Gaussian noise field with power spectral density σ_η^2, determine the region D_1.

B. Detection Problem Solution

The wavelet transform of $f_\eta(x, y)$ is

$$W_{2^j}^i f_\eta(n, m) = \begin{cases} W_{2^j}^i f_o(n, m) + W_{2^j}^i \eta(n, m) & \text{if } (i, j, n, m) \in D_1 \\ W_{2^j}^i \eta(n, m) & \text{otherwise,} \end{cases} \quad (8)$$

where $W_{2^j}^i \eta(n, m)$ is the wavelet transform of $\eta(x, y)$. Note that $W_{2^j}^i f_\eta(n, m)$ is a zero-mean uncorrelated Gaussian random sequence whose variance is $\sigma_\eta^2 + \sigma_{f_o}^2$ for (i, j, n, m) inside D_1 and σ_η^2 for (i, j, n, m) outside D_1. Therefore, the decision of whether a point $(i, j, m, n) \in D_1$ decouples from similar decisions for other points. The solution of the problem is the test [10]

$$|W_{2^j}^i f_\eta(n, m)| \underset{\in D_1}{\overset{\in D_1}{\gtrless}} \nu \quad (9)$$

where ν is a threshold, determined using, e.g., a Neyman-Pearson criterion. This means that we can decide whether (i, j, n, m) is in

D_l simply by thresholding the absolute value of the wavelet transform of the noisy image at that point. When the whiteness or orthonormality assumptions are relaxed, the threshold test described above is no longer guaranteed to be optimal. However, the threshold test still seems to be the "natural" approach.

IV. Image Reconstruction under Wavelet Constraints

For clarity of presentation, we first define and solve the problem of image reconstruction under wavelet constraints at a single scale and on a single subwavelet. We then generalize to multiple scales and several subwavelets.

A. Constrained Wavelet Coefficients on a Single Scale

Suppose that we are given noisy observations $p_\eta(m, n) = p(m, n) + \eta(m, n)$ of the projections $p(m, n)$, where $\eta(m, n)$ is a zero-mean Gaussian noise field which is uncorrelated in the angular variable and correlated in the radial variable with autocorrelation $R_\eta(k) = E[\eta(m, n)\eta(m + k, n)]$. Suppose also that the wavelet coefficients of the actual image $\mu_a(i, j)$ at translations $\{(i_c, j_c), 1 \le c \le C\}$ on the finest scale 2^1, and with respect to the first subwavelet $\psi^1(x, y)$ are known or estimated (see Section III) to be zero

$$W^1_{2^1}\mu_a(i_c, j_c) = \sum_i \sum_j \mu_a(i, j) g^1(2i_c - i, 2j_c - j) = 0, \quad 1 \le c \le C. \tag{10}$$

Our goal is to compute the image $\tilde{\mu}(i, j)$ such that
1) $\tilde{\mu}(i, j)$ satisfies the wavelet constraints; and
2) $E\{\Sigma_i \Sigma_j (\mu_a(i, j) - \tilde{\mu}(i, j))^2\}$ is minimized.

Let $\mu_\eta(i, j) = \mathcal{R}_d^{-1}\{p_\eta(m, n)\}$, and $\epsilon(i, j) = \mathcal{R}_d^{-1}\{\eta(m, n)\}$ so that $\mu_\eta(i, j) = \mu_a(i, j) + \epsilon(i, j)$. To solve the problem, we first estimate $\epsilon(i, j)$ using the given constraints and then we subtract the noise estimates from the noisy image $\mu_\eta(i, j)$. Since $\eta(m, n)$ is zero-mean Gaussian, $\epsilon(i, j)$ is also zero-mean Gaussian, and the solution is the linear minimum mean square (LMMSE) estimate. The constraints (10) can be written as $\{W^1_{2^1}\epsilon(i_c, j_c) = W^1_{2^1}\mu_\eta(i_c, j_c), 1 \le c \le C\}$. The goal is to compute the MMSE of $\epsilon(i, j)$ subject to these constraints.

Let $R_\epsilon(i, j)$ be the autocovariance of $\epsilon(i, j)$, let $R_{W^1_{2^1\epsilon}, W^1_{2^1\epsilon}}(2(i - k), 2(j - l)) = E[W^1_{2^1}\epsilon(i, j) W^1_{2^1}\epsilon(k, l)]$, and let $R_{(W^1_{2^1\epsilon}), (\epsilon)}(2i - k, 2j - l) = E[W^1_{2^1}\epsilon(i, j)\epsilon(k, l)]$. Using a discrete variable extension of the results of [11], it can easily be shown that

$$R_\epsilon(i, j) = \mathcal{R}_d^{-1}\{q(m) * R_\eta(m)\}. \tag{11}$$

The functions $R_{W^1_{2^1\epsilon}, W^1_{2^1\epsilon}}(2(i - k), 2(j - l))$ and $R_{(W^1_{2^1\epsilon}), (\epsilon)}(2i - k, 2j - l)$ are computed as

$$\begin{aligned} R_{W^1_{2^1\epsilon}, W^1_{2^1\epsilon}}(2(i - k), 2(j - l)) &= \sum_m \sum_n \sum_s \sum_t R_\epsilon(m, n) \\ &\quad \cdot g^1(s, t) g^1(s + 2(i - k) - m, \\ &\quad t + 2(j - l) - n) \\ R_{(W^1_{2^1\epsilon}), (\epsilon)}(2i - k, 2j - l) &= \sum_m \sum_n R_\epsilon(m, n) \\ &\quad \cdot g^1(2i - k - m, 2j - l - n). \end{aligned} \tag{12}$$

The LMMSE estimate of $\epsilon(i, j)$ is

$$\hat{\epsilon}(i, j) = \sum_{c=1}^{c=C} \beta_c R_{(W^1_{2^1\epsilon}), (\epsilon)}(2i_c - i, 2j_c - j) \tag{13}$$

where $\{\beta_c, 1 \le c \le C\}$ are computed by solving the $C \times C$ linear system of equations

$$[\beta_1, \cdots, \beta_C]M = [W^1_{2^1}\mu_\eta(i_1, j_1), \cdots, W^1_{2^1}\mu_\eta(i_C, j_C)] \tag{14}$$

and the (u, v)th entry of the $C \times C$ matrix M is $R_{W^1_{2^1\epsilon}, W^1_{2^1\epsilon}}(2(i_v - i_u), 2(j_v - j_u))$.

Finally, the constrained reconstructed image is $\tilde{\mu}(i, j) = \mu_\eta(i, j) - \hat{\epsilon}(i, j)$.

B. Constrained Wavelet Coefficients on Multiple Scales

We now generalize the results of the previous subsection to multiple scales and more than one subwavelet. For notational simplicity, we assume that two subwavelets are used; generalization to more than two subwavelets should be apparent.

Assume that the wavelet transform of $\mu_a(i, j)$ is known to be zero at various points on L different scales. We index each of these points by an integer $c_1(l)$ or $c_2(l)$ where l denotes the scale and the subscript denotes the subwavelet used. Similarly to (10), these constraints can be written as

$$\begin{aligned} W^1_{2^l}\epsilon(i_{c_1(l)}, j_{c_1(l)}) &= W^1_{2^l}\mu_\eta(i_{c_1(l)}, j_{c_1(l)}), \\ &\qquad 1 \le c_1(l) \le C_1(l), \quad 1 \le l \le L \\ W^2_{2^l}\epsilon(i_{c_2(l)}, j_{c_2(l)}) &= W^2_{2^l}\mu_\eta(i_{c_2(l)}, j_{c_2(l)}), \\ &\qquad 1 \le c_2(l) \le C_2(l), \quad 1 \le l \le L \end{aligned} \tag{15}$$

where the superscripts denote the subwavelet being used.

Having established this notation, the arguments used in Section IV-A lead to the formula

$$\begin{aligned} \hat{\epsilon}(i, j) = &\sum_{l=1}^{L} \sum_{c_1(l)=1}^{C_1(l)} \beta_{c_1(l)} R_{(W^1_{2^l\epsilon}), (\epsilon)}(2^l i_{c_1(l)} - i, 2^l j_{c_1(l)} - j) \\ &+ \sum_{l=1}^{L} \sum_{c_2(l)=1}^{C_2(l)} \beta_{c_2(l)} R_{(W^2_{2^l\epsilon}), (\epsilon)}(2^l i_{c_2(l)} - i, 2^l j_{c_2(l)} - j) \end{aligned} \tag{16}$$

where the β are again computed by solving a linear system of equations. The right-hand side of the system is a vector of the known values of $W^1_{2^l}\mu_\eta(i_{c_1(l)}, j_{c_1(l)})$ and $W^2_{2^l}\mu_\eta(i_{c_2(l)}, j_{c_2(l)})$, which are again the wavelet coefficients known to be zero in the *noiseless* image. The system matrix M consists of L^2 submatrices $M_{l,m}$, each of which contains the cross-covariance between wavelet coefficients of the noise at scales l and m. M and $M_{l,m}$ have the forms

$$M = \begin{bmatrix} M_{1,1} & M_{1,2} & \cdots & M_{1,L} \\ M_{2,1} & M_{2,2} & \cdots & M_{2,L} \\ \vdots & \vdots & & \vdots \\ M_{L,1} & M_{L,2} & \cdots & M_{L,L} \end{bmatrix}$$

$$M_{l,m} = \begin{bmatrix} M_{l,m}(1, 1) & M_{l,m}(1, 2) \\ M_{l,m}(2, 1) & M_{l,m}(2, 2) \end{bmatrix} \tag{17}$$

where $M_{l,m}(1, 1)$ contains the cross-covariance of noise wavelet coefficients computed with respect to the first subwavelet. The $(c_1(l), c_1(m))$th entry of $M_{l,m}(1, 1)$ is $R_{W^1_{2^l\epsilon}, W^1_{2^m\epsilon}}(2^l i_{c_1(l)} - 2^m i_{c_1(m)}, 2^l j_{c_1(l)} - 2^m j_{c_1(m)})$. Entries of $M_{l,m}(1, 2)$, and $M_{l,m}(2, 2)$ are defined similarly; the cross-covariance between subwavelets 1 and 2 are used for $M_{l,m}(1, 2)$, and cross-covariances computed with respect to the second subwavelet is used for $M_{l,m}(2, 2)$. Again, the reconstructed image is $\tilde{\mu}(i, j) = \mu_\eta(i, j) - \hat{\epsilon}(i, j)$.

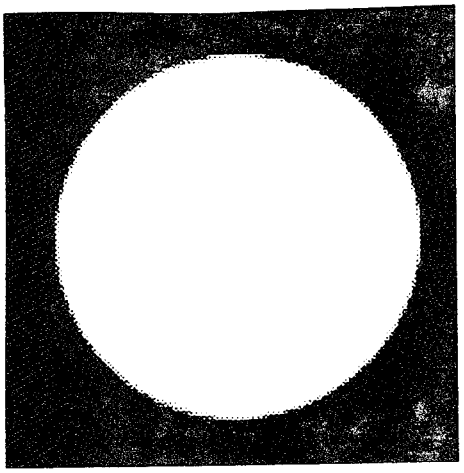


Fig. 1. The reconstruction of the disk used in Example 1 from noiseless projections.

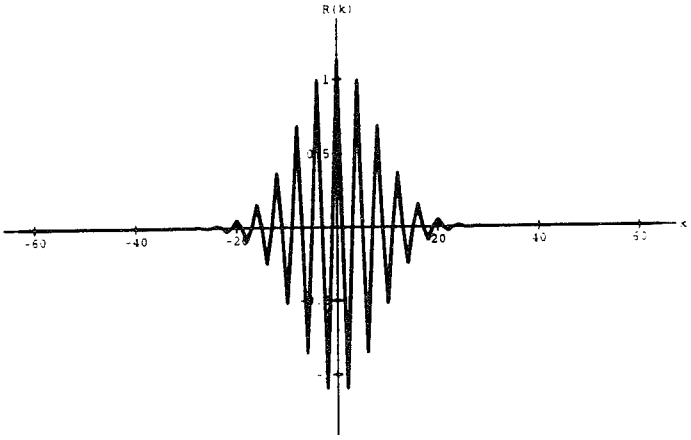


Fig. 2. The autocorrelation of the noise added to each projection, as a function of the radial distance.

V. Numerical Results

We present four numerical examples which illustrate the above procedures.

Example 1: The noiseless image used in this example is a disk of value 1.00 and radius 0.81 in a background of value 0. Projections of the disk are computed over 128 angles and 128 lines in each angle. The reconstruction from noiseless projections is shown in Fig. 1. The noise added to the projections is obtained by passing zero-mean white Gaussian noise with variance 0.01 through a filter whose discrete-time Fourier transform is $(\sin(w))^{32}$. The autocovariance of the noise is shown in Fig. 2, and the 100×100 image obtained from the noisy projections using FBP is shown in Fig. 3.

The wavelets we use are two subwavelets of the Haar basis, which can be regarded as difference operators in the x and y directions (the third Haar subwavelet, which can be regarded as a difference operator in the diagonal direction, is not used). We constrain the two finest-scale wavelet coefficients to be zero in a 15×55 rectangular area R_0 inside the disk. We use (16) to estimate the noise $\epsilon(i, j)$, where the matrix M is given by (17).

The MMSE image is shown in Fig. 4. The first row in Table I discusses the average performance of our procedure for 10 different noise realizations for this example. The area obtained by enlarging R_0 by 20 pixels in every direction is denoted as R_1; this is roughly the area in which we expect improvement, due to the support of the filter. The whole image is denoted by R_t. From Table I, we see

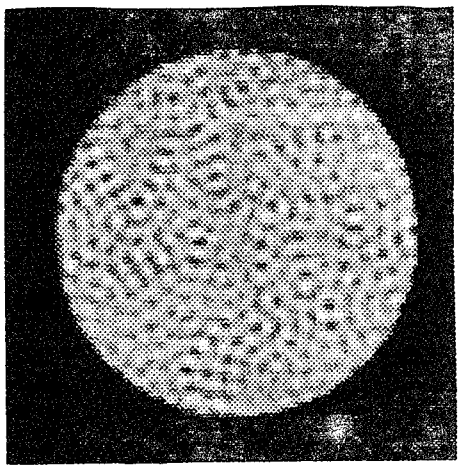


Fig. 3. The noisy image for Example 1, obtained from the noisy projections by FBP.

TABLE I

	Noise Power in			% Improvement in	
	R_0	R_1	R_t	$R_1 - R_0$	$R_t - R_0$
Noisy Image	5.20	28.48	53.35		
Example 1	0.082	20.30	45.15	13.15	6.40
Example 2	2.475	24.48	49.78	5.47	1.75

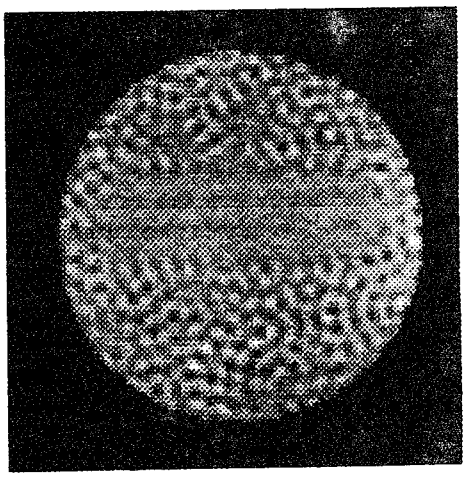


Fig. 4. The MMSE image obtained by constraining the two finest scale wavelet coefficients in R_0 to 0.

that the noise in R_0 is almost completely eliminated. We also find that, compared to the unprocessed noisy image, the noise powers in the regions $R_1 - R_0$ and $R_t - R_0$, *in which we do not have any wavelet constraints*, are reduced by 13.1 and 6.4%, respectively. This shows that constraining wavelet coefficients in a given region *improves the reconstruction in other regions*. This is because the noise ϵ in the reconstructed image is nonwhite, due to the fact that the Radon transform is nonunitary and the additive noise is nonwhite in the projections is nonwhite.

Example 2: In the second example, the original image is not flat in R_0, but varies smoothly in that region. The noiseless image, which is the union of a disk and an exponential, is shown in Fig. 5. The additive noise is the same as in Example 1 and the noisy reconstruction is shown in Fig. 6. The wavelet constraints used are the same as in Example 1.

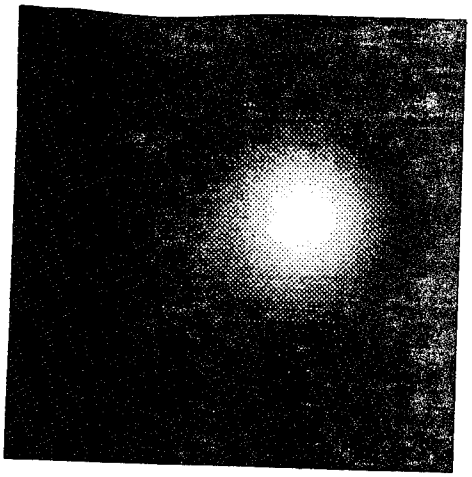


Fig. 5. The reconstruction of the image used in Examples 2 and 3 from noiseless projections.

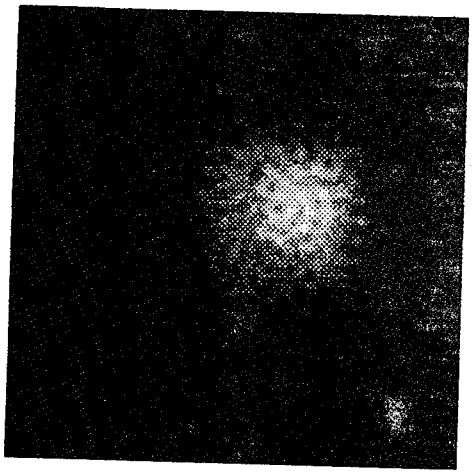


Fig. 6. The noisy image for Examples 2 and 3, obtained from the noisy projections by FBP.

The resulting image is shown in Fig. 7. From Table I, we see that the noise in R_0 is not completely eliminated. This is mostly because we are using the Haar wavelet, which is not smooth and has only one vanishing moment. The noiseless image has small, but not negligible, high-resolution components, which results in nonzero fine-scale wavelet coefficients when the Haar basis is used. This suggests using a different wavelet, which we now do.

Example 3: In the third example, we use the same image and noise as in Example 2, but we use two scales of the six-coefficient Daubechies wavelet [3], which has three vanishing moments. The resulting MMSE image is shown in Fig. 8. The noise power inside R_0, which was reduced to 2.5 in Example 2, is now further reduced to 0.77 with this choice of a smoother wavelet.

Example 4: In this example, we use the Shepp–Logan head phantom [12] as our noiseless image, which is shown in Fig. 9. The autocorrelation of the noise is the same as previous examples, and the noise variance is 4×10^{-6}. The 128×128 noisy image is shown in Fig. 10. We use the Haar wavelet, and constrain the two finest-scale wavelet coefficients to be zero over a region D of the image. The region D is obtained by the thresholding approach over a region D' in the center of the image. First, the second-finest wavelet coefficients inside D' are set to zero whenever their absolute value is below a threshold. The region which will be affected by the above operation is called D''. Then, inside D'', another

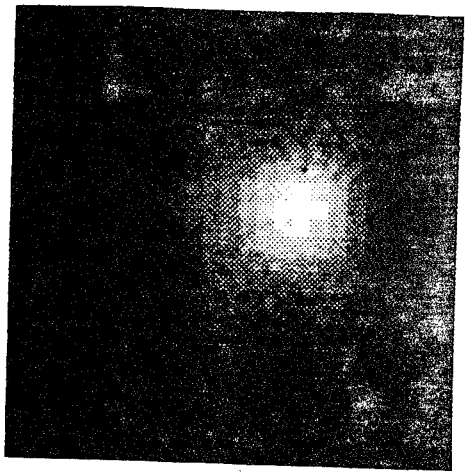


Fig. 7. The MMSE image obtained by constraining the two finest-scale Haar basis wavelet coefficients in R_0 to 0; Example 2.

Fig. 8. The MMSE image obtained by constraining the two finest-scale wavelet coefficients in R_0 to 0; the wavelet basis function is the 6 coefficient Daubechies wavelet; Example 3.

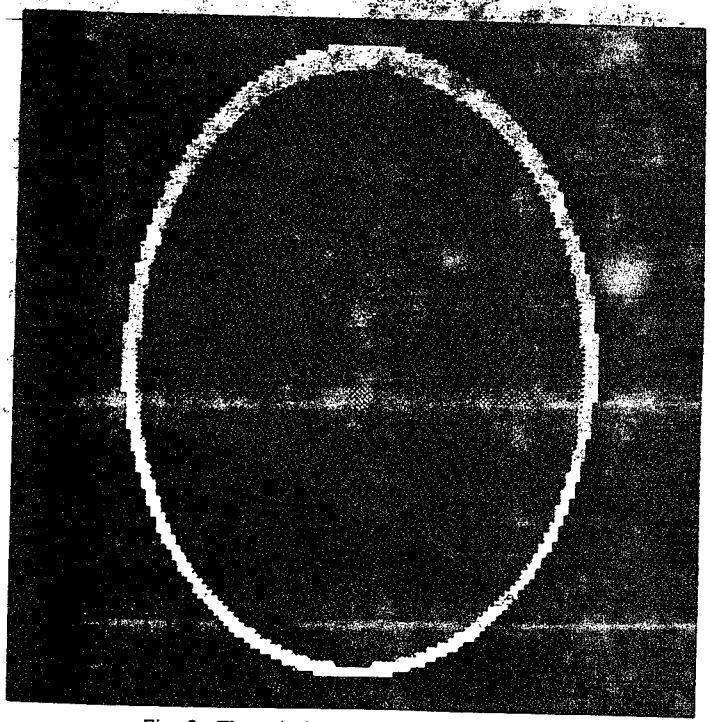


Fig. 9. The noiseless Shepp–Logan phantom.

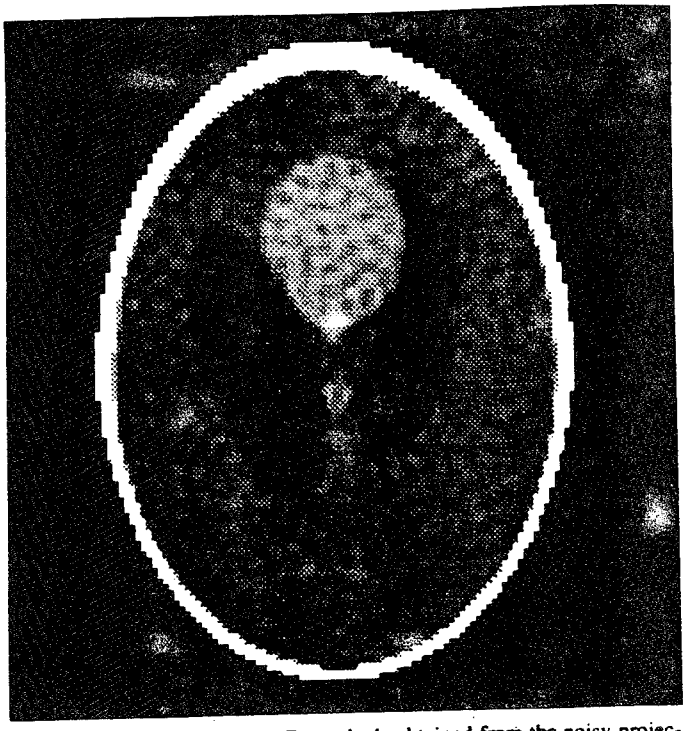


Fig. 10. The noisy image for Example 4, obtained from the noisy projections of the Shepp–Logan phantom.

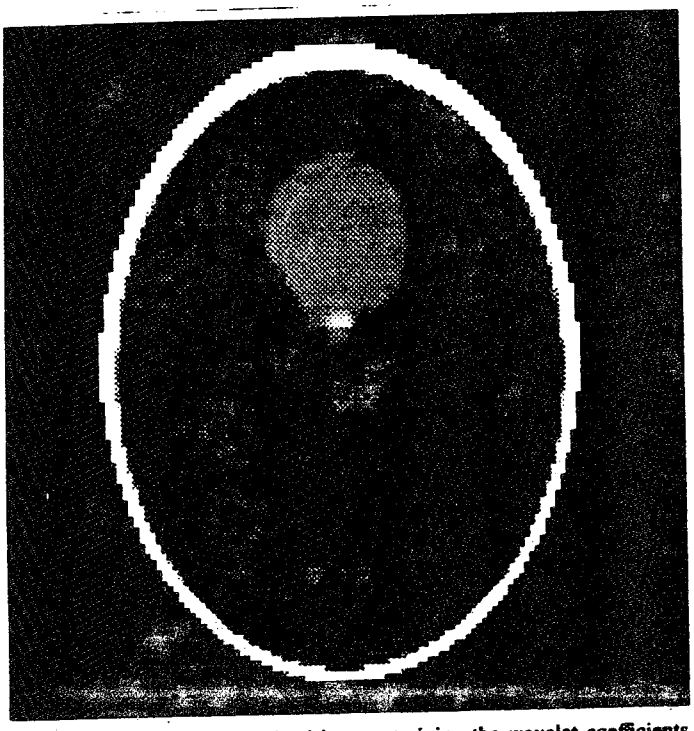


Fig. 11. MMSE image obtained by constraining the wavelet coefficients of the noisy image; Example 4.

threshold is used to set the finest-scale coefficients to zero. The resulting MMSE image is shown in Fig. 11. The noise power has been reduced by 20.3% in the whole image, while still preserving the edges.

References

[1] S. R. Deans, *The Radon Transform and Some of its Applications*, New York: Wiley, 1983.

[2] S. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-11, pp. 674–693, 1989.

[3] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *IEEE Trans. Inform. Theory*, vol. 36, pp. 961–1005, 1990.

[4] B. E. A. Saleh and N. S. Subotic, "Time-variant filtering of signals in the mixed time-frequency domain," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, pp. 1479–1487, 1985.

[5] T. E. Koczwara and D. L. Jones, "On mask selection for time-varying filtering using the Wigner distribution," in *Proc. ICASSP*, Albuquerque, NM, 1990, pp. 2487–2490.

[6] M. Bikdash and K. B. Yor, "Linear shift varying filtering of nonstationary chirp signals," in *Proc. ICASSP*, New York, 1988, pp. 428–432.

[7] B. Boashash and L. B. White, "Instantaneous frequency estimation and automatic time-varying filtering," in *Proc. ICASSP*, Albuquerque, NM, 1990, pp. 1221–1224.

[8] J. Jeong and W. J. Williams, "Time-varying filtering and signal synthesis," in *Time-Frequency Signal Analysis*, B. Boashash, Ed. Melbourne: Longman and Cheshire, 1991.

[9] G. F. Boudreaux-Bartels and T. W. Parks, "Time-varying filtering and signal estimation using Wigner distribution synthesis techniques," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 34, pp. 442–451, 1988.

[10] H. L. Van Trees, *Detection, Estimation, and Modulation Theory*, New York: Wiley, 1968.

[11] A. K. Jain and S. Ansari, "Radon transform theory for random fields and optimum image reconstruction from noisy projections," in *Proc. ICASSP*, San Diego, CA, 1984, pp. 12A.7.1–12A.7.4.

[12] L. A. Shepp and B. F. Logan, "The Fourier reconstruction of a head section," *IEEE Trans. Nucl. Sci.*, vol. NS-21, pp. 21–42, 1974.

Multiresolution Representations Using the Auto-Correlation Functions of Compactly Supported Wavelets

Naoki Saito and Gregory Beylkin

***Abstract*—We propose a shift-invariant multiresolution representation of signals or images using dilations and translations of the auto-correlation functions of compactly supported wavelets. Although these functions do not form an orthonormal basis, their properties make them useful for signal and image analysis. Unlike wavelet-based orthonormal representations, our representation has 1) symmetric analyzing functions, 2) shift-invariance, 3) associated iterative interpolation schemes, and 4) a simple algorithm for finding the locations of the multiscale edges as zero-crossings.**

We also develop a noniterative method for reconstructing signals from their zero-crossings (and slopes at these zero-crossings) in our representation. This method reduces the reconstruction problem to that of solving a system of linear algebraic equations.

Manuscript received February 3, 1992; revised June 3, 1993. The Guest Editor coordinating the review of this paper and approving it for publication was Prof. Martin Vetterli.

N. Saito is with Schlumberger-Doll Research, Ridgefield, CT 06877 and the Department of Mathematics, Yale University, New Haven, CT 06250.

G. Beylkin is with the Program in Applied Mathematics, University of Colorado at Boulder, Boulder, CO 80309-0526.

IEEE Log Number 9212194.

1053-587X/93$03.00 © 1993 IEEE

APPENDIX J2

B. Sahiner and A.E. Yagle, "On the Use of Wavelets in Inverting the Radon Transform," IEEE 1992 Medical Imaging Conference, Orlando, FL, Oct. 25-31, 1992, pp. 1129-31.

This is the conference paper version of Appendix J1.

ON THE USE OF WAVELETS IN INVERTING THE RADON TRANSFORM

Berkman Sahiner and Andrew E. Yagle *
Department of Electrical Engineering and Computer Science
The University of Michigan, Ann Arbor, Michigan 48109-2122

Abstract

In this paper, we first present some new results on constrained image reconstruction. Given constraints on pixel values and the statistics of additive noise in the projections, we show how to compute the minimum mean-square estimate of the reconstructed image. Second, we present some results on the use of the wavelet transform to perform spatially-varying filtering of the image and show how noise can be suppressed in flat areas of the image. Third, we combine the previous results into a new constrained image reconstruction procedure, in which image constraints are applied in the wavelet domain. The new procedure improves the reconstructed image not only in locations where wavelet constraints are applied, but also in other regions.

I. Introduction

In many problems arising in fields such as medical imaging, non-destructive testing, and geophysics [2], one needs to reconstruct a two-dimensional object or image from its projections, which amounts to computing the inverse Radon transform. The inverse Radon transform is ill-conditioned; if it is regularized by using a low-pass filter, then high-resolution features in the image are smoothed or lost. It is desirable to reduce the noise energy in the reconstructed image over regions where high-resolution features are not present, by using spatially-varying filtering.

In this paper, we use wavelets to perform this desired spatially-varying filtering. We constrain fine-scale wavelet coefficients to zero in certain regions in the wavelet domain, effecting localized low-pass filtering. To develop the necessary algorithm, we first discuss the problem of computing the minimum mean-square error estimate of the reconstructed image which satisfies given constraints on some pixel values. This leads us to an algorithm which operates directly on the reconstructed image. We then modify the results of this algorithm and apply it to the more realistic problem of constraining the wavelet coefficients.

II. Constrained Image Reconstruction from Projections

Basic Problem: The Radon transform inversion problem is the basic problem of image reconstruction from projections. Let $\mu(x,y)$ and $p(r,\theta)$ denote an image and its Radon transform, (projections) respectively. The most common procedure for obtaining $\mu(x,y)$ from $p(r,\theta)$ is filtered backprojection (FBP), in which the projections are first filtered to yield $s(r,\theta) = p(r,\theta) * q(r)$, where the Fourier transform of the filter $q(r)$ approximates $|w|$. The image $\mu(x,y)$ is then obtained by backprojecting $s(r,\theta)$. In practical problems, we have only samples of $p(r,\theta)$, in which case samples of the image $\mu(i,j)$ are obtained by a discretization of the continuous FBP method:

$$\mu(i,j) = \mathcal{R}_d^{-1}\{p(m,n)\} = \frac{1}{2N}\sum_{n=0}^{N-1}\sum_{m=-\infty}^{\infty}\sum_{l=-\infty}^{\infty} q(l-m)p(m,n)h(i\cos n\Delta + j\sin n\Delta - l) \quad (1)$$

where N is the total number of views, $\Delta = \pi/N$, $p(m,n)$ is the m^{th} projection in the n^{th} view, $h(x)$ is an interpolation function and $\mathcal{R}_d^{-1}$ is the discrete version of the Inverse Radon transform operator. Note that $q(\cdot)$ may be chosen to filter noise in the projections; however it will also smooth the entire reconstructed image.

Constrained Minimum Mean Square Estimate: Suppose that we are given noisy observations $p_\eta(m,n) = p(m,n) + \eta(m,n)$ of the projections $p(m,n)$, where $\eta(m,n)$ is a zero-mean Gaussian noise field, uncorrelated between views (i.e., in the angular variable), but correlated within a view with autocorrelation $R_\eta(m) = E[\eta(k,n)\eta(m+k,n)]$. Suppose that we are also given the value of the actual image $\mu_a(i_c,j_c)$ at C points. The problem addressed in this section is to find the image $\tilde{\mu}(i,j)$ such that:

1. $\tilde{\mu}(i,j)$ satisfies the constrained image values $\{\mu_a(i_c,j_c) = K_c, 1 \le c \le C\}$; and
2. $E\{\sum_i \sum_j (\mu_a(i,j) - \tilde{\mu}(i,j))^2\}$ is minimized,

To solve this problem, we first compute the noisy image $\mu_\eta(i,j)$ from the noisy projections $p_\eta(m,n)$. Second, we determine the noise values at the constrained pixels (i_c,j_c) by subtracting the actual pixel values K_c from the noisy image pixel values $\mu_\eta(i_c,j_c)$. Third, we estimate the noise values at other pixels from the known noise values at the constrained pixels. Finally, we subtract the noise estimates from the noisy image $\mu_\eta(i,j)$.

Specifically, let $\epsilon(i,j) = \mathcal{R}_d^{-1}\{\eta(m,n)\}$. Then, the problem is to compute the minimum mean-square estimate $\hat{\epsilon}(i,j)$ given the known values $\{\epsilon(i_c,j_c) = \mu_\eta(i_c,j_c) - K_c, 1 \le c \le C\}$. Since $\eta(m,n)$ is zero-mean Gaussian, $\epsilon(i,j)$ is also zero mean Gaussian, and the solution is the linear minimum mean square (LMMSE) estimate.

To compute the LMMSE, we first need the autocovariance $R_\epsilon(i,j)$ of the noise $\epsilon(i,j)$. It can be shown [1] that the autocovariance of the noise in the reconstructed image is

*This work was supported by the Office of Naval Research under grant #N00014-90-J-1897

0-7803-0883-2/93/$3.00 ©1993 IEEE

$$R_\epsilon(i,j) = E\{\epsilon(i',j')\epsilon(i+i',j+j')\} \simeq \mathcal{R}_d^{-1}\{q(m) * R_\eta(m)\} \quad (2)$$

The LMMSE $\hat{\epsilon}(i,j)$ of $\epsilon(i,j)$ is found as [1]

$$\hat{\epsilon}(i,j) = \sum_{c=1}^{C} \beta_c R_\epsilon(i-i_c, j-j_c). \quad (3)$$

where the β_c are computed by solving the $C \times C$ linear system of equations

$$[\beta_1, \ldots, \beta_C]\, M = [\mu_\eta(i_1,j_1) - K_1, \ldots, \mu_\eta(i_C,j_C) - K_C] \quad (4)$$

and the $(u,v)^{th}$ element of the matrix M is $R_\epsilon(i_v - i_u, j_v - j_u)$. The final LMMSE $\tilde{\mu}(i,j)$ of the image is then $\tilde{\mu}(i,j) = \mu_\eta(i,j) - \hat{\epsilon}(i,j)$.

Note that when the additive noise on the projections is white, $R_\epsilon(i,j)$ is equal to $t(i,j) \doteq \mathcal{R}_d^{-1}\{q(m)\}$, and each pixel constraint affects $\tilde{\mu}(i,j)$ not only at the constrained pixels $(i_1,j_1), \ldots, (i_C,j_C)$ but also in the vicinity of each point. The size of this vicinity is the size of support of $t(i,j)$. Hence constraining some pixel values also affects other pixel values. Depending on the autocovariance of $\eta(m,n)$, the support of $R_\epsilon(i,j)$ may become very significant.

III. The Wavelet Transform

The wavelet transform is a time-frequency tool that has good localizing properties. The 2-D continuous wavelet transform $W_{2^j} f(x,y)$ of a 2-D, square-integrable function $f(x,y)$ is defined as the convolution of $f(x,y)$ with dilations of a wavelet basis function $\psi(x,y)$ at different scales

$$W_{2^j} f(x,y) = f(x,y) * *\psi_{2^j}(x,y)$$
$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(z_1,z_2)\psi_{2^j}(x-z_1, y-z_2)\, dz_1 dz_2, \quad (5)$$

where $\psi_{2^j}(x,y) = 2^{-j}\psi\left(2^{-j}x, 2^{-j}y\right)$, and $\psi(x,y)$ is the wavelet basis function.

The wavelet transform of an image can be computed directly from its Radon transform. Let $W_{2^j}\mu(x,y)$ be the wavelet transform of an image $\mu(x,y)$ and let $\Psi(r,\theta) = \mathcal{R}\{\psi(x,y)\}$ denote the Radon transform of the wavelet basis function $\psi(x,y)$. Using the scaling and convolution properties of the Radon transform [2], we obtain

$$W_{2^j}\mu(x,y) = \mu(x,y) * *\psi_{2^j}(x,y)$$
$$= \mathcal{R}^{-1}\{p(r,\theta) * \Psi(2^{-j}r, \theta)\}. \quad (6)$$

Thus the wavelet transform of the reconstructed image can be found by first computing a wavelet-like transform of each projection, and then taking the inverse Radon transform.

For image processing applications, several sub-wavelets are used, each having a specific orientation [3]. For example, for edge detections applications, two wavelet basis functions $\psi^1(x,y)$ and $\psi^2(x,y)$ are used, and two sets of wavelet coefficients are computed at each scale:

$$W^1_{2^j} f(x,y) = f(x,y) * *\psi^1_{2^j}(x,y),$$
$$W^2_{2^j} f(x,y) = f(x,y) * *\psi^2_{2^j}(x,y). \quad (7)$$

The continuous wavelet transform (5) is redundant and can be sampled to yield the discrete wavelet transform, defined as

$$W_{2^j} f(n,m) = W_{2^j} f(x,y)|_{x=2^j n, y=2^j m} \quad (8)$$

IV. Multiscale Filtering

The idea of filtering a signal in time-frequency space has been discussed and employed for the Wigner distribution and short-time Fourier transform in [4] and [5]. In this paper, we perform a simple version of multiscale filtering by setting the wavelet coefficients of the signal to zero in regions of the wavelet transform domain where the signal energy is known or estimated to be much smaller than the noise energy. In particular, we set the fine-scale wavelet coefficients to zero in the regions of the image which are free of high-resolution features. Since fine-scale wavelet transform components represent localized high-resolution features of the image, windowing these to zero effectively smoothes the image, much as lowpass filtering does. The advantage of using wavelets is that this can be done on a localized basis, smoothing in some areas while leaving other areas (such as edges) alone.

The region in the wavelet space where the fine-scale wavelet coefficients are set to zero can be determined from either a priori knowledge of image features or statistical knowledge about the image and noise.

A priori knowledge of image features: Often in image reconstruction problems there is considerable a priori knowledge about the image. For example, it may be known, a priori, that there are no high-resolution features in some region D of the image. D may be known to represent a flat or slowly-varying part of the image, or D may be known to be free of edges. Since fine-scale wavelet transform components represent localized high-resolution features of the image, we can window these to zero in the region D, and cancel some of the noise energy in D without degrading the image.

Statistical knowledge about the image and noise: If there is no a priori knowledge of image features, a thresholding approach may be used to set some wavelet coefficients to zero. The idea is to eliminate noise where it is possible to do so without significantly degrading the image. Since the wavelet transform is being used, this can be done on a localized basis. In this paper, we set the fine-scale wavelet coefficients to zero in regions where they fall below a threshold, a method that was previously applied to other time-frequency distributions [5]. In [1], we describe a situation where this thresholding approach is optimal.

V. Complete Procedure and Numerical Results

The results of Section II are not very useful in themselves, since it is most unlikely that we have a priori knowledge of the actual values of the image. However, we will have constraints on the values (all known to be zero) of some wavelet coefficients. In this section, we modify the results

of Section II so that constraints on wavelet coefficients are used to improve the image, and give an example.

We use constraints on two sub-wavelets in this paper, and assume that the discrete wavelet transform $W^1_{2^l}\mu_a(i,j)$ and $W^2_{2^l}\mu_a(i,j)$ of $\mu_a(i,j)$ (defined by (5) and (8)) is known or estimated to be zero at various points on L scales. We index each of these points by an integer $c_1(l)$ or $c_2(l)$ where l denotes the scale and the subscript denotes the sub-wavelet used. It is easily shown [1] that the LMMSE noise is given by

$$\hat{\epsilon}(i,j) = \sum_{l=1}^{L} \sum_{c_1(l)=1}^{C_1(l)} \beta_{c_1(l)} R_{(W^1_{2^l}\epsilon),(\epsilon)}(i - i_{c_1(l)}, j - j_{c_1(l)})$$

$$+ \sum_{l=1}^{L} \sum_{c_2(l)=1}^{C_2(l)} \beta_{c_2(l)} R_{(W^2_{2^l}\epsilon),(\epsilon)}(i - i_{c_2(l)}, j - j_{c_2(l)}), \qquad (9)$$

where $R_{(W_{2^l}\epsilon),(\epsilon)}$ is the cross-covariance between $\epsilon(i,j)$ and $W_{2^l}\epsilon(i,j)$ and the β are again computed by solving a linear system of equations similar to (4), for details see [1].

Example: We use the Shepp-Logan phantom as our noiseless image, which is shown in Fig. 1. The noise added to the projections is obtained by passing zero-mean white Gaussian noise with variance 4×10^{-6} through a filter whose discrete-time Fourier transform is $(sin(w))^{32}$. The 128×128 image obtained from the noisy projections using FBP is shown in Fig. 2. The wavelets that we use are 2-D versions of the 1-D Haar wavelet, which are defined as $\psi^1(x,y) = sq(x,y) + sq(x-1,y) - sq(x,y-1) - sq(x-1,y-1)$ and $\psi^2(x,y) = \psi^1(y,x)$; where $sq(x,y)$ is unity over the square region $[0,1] \times [0,1]$ and zero elsewhere. We constrain the two finest-scale wavelet coefficients to be zero over a region D of the image. The region D is obtained by thresholding the second-finest wavelet coefficients of the noisy image in a region D' in the center of the image. The resulting MMSE image is shown in Fig. 3. The noise power has been decreased by 20.3% in the whole image and by 10.1% outside the regions where wavelet coefficients are constrained.

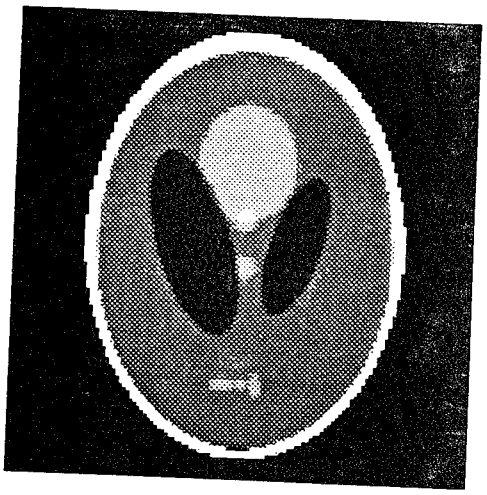


Figure 1: Noiseless Phantom

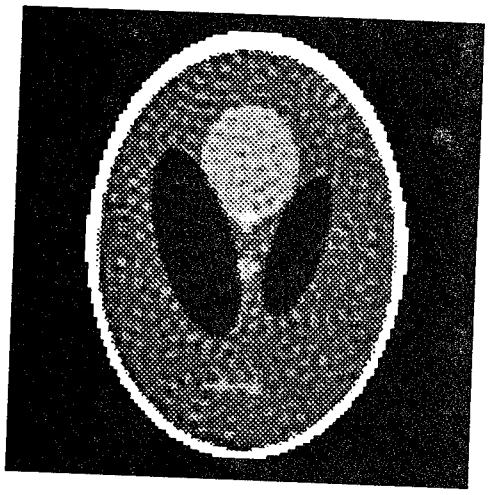


Figure 2: Phantom reconstructed from noisy projections

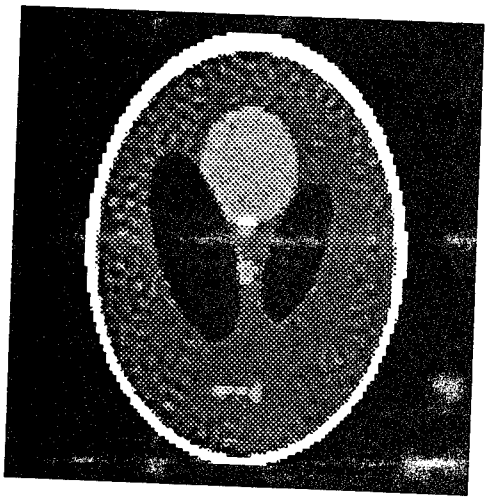


Figure 3: MMSE image obtained by constraining the wavelet coefficients of the noisy image.

References

[1] B. Sahiner and A.E. Yagle, "Constrained Image Reconstruction from Projections Using the Wavelet Transform," submitted to IEEE Trans. Sig. Proc., Special Issue on Wavelet Transforms.

[2] S. R. Deans, *The Radon Transform and Some of its Applications*. New York: Wiley, 1983.

[3] S. Mallat, "Multifrequency channel decompositions of images and wavelet models," IEEE Trans. Acoust., Speech, Sig. Proc., vol. ASSP-37, pp. 2091-2110, 1989.

[4] B. E. A. Saleh and N. S. Subotic, "Time-variant filtering of signals in the mixed time-frequency domain," IEEE Trans. Acoust., Speech, Sig. Proc., vol. ASSP-33, pp. 1479-1487, 1985.

[5] T. E. Koczwara and D. L. Jones, "On mask selection for time-varying filtering using the Wigner Distribution, in Proc. ICASSP 1990, pp. 2487-2490.

APPENDIX K1

B. Sahiner and A.E. Yagle, "Time-Frequency Distribution Inversion of the Radon Transform," to appear in IEEE Trans. Image Proc. 2(4), October 1993.

This paper performs a time-frequency analysis of the projection data in the inverse Radon transform problem. Regions in time-frequency space in which the distribution strength is below a threshold are assumed to be due to noise, and are set to zero. This has the effect of filtering noise out of time-frequency regions in which the signal strength is small, and leaving the noise in where the signal strength is large. The resulting time-frequency distribution is then projected to find the nearest feasible signal solution, which is then backprojected. This reduces noise in the reconstructed image while maintaining sharpness of image features.

Fig. 9. Top left: input frames with 1-D motion blurs. Top right: input frames with 2-D blurs. Bottom left: reconstructed image using input frames from the top left with $\lambda = 1.0$. Bottom right: reconstructed image using input frames from the top right with $\lambda = 1.0$.

has been developed. Due to the deblurring process, the high-resolution reconstruction is not stable. By using a recursive scheme with an adaptively updated regularization parameter, an effective high-resolution reconstruction has been obtained. A new input image frame can be incorporated to the reconstruction in a very efficient way without repeating the whole computation. The computation can be implemented in a highly parallel scheme since all the DFT components of the reconstructed image are computed independently. The regularization parameters should be chosen appropriately, balancing the deblurring, high-resolution restoration and noise amplifications. In order to further improve the results, the batch mode iterative computation (for fixed number of input frames) may be incorporated between the recursive reconstructions with new input frames.

REFERENCES

[1] S. P. Kim, N. K. Bose, Valenzuela, "Recursive reconstruction of high resolution image from noisy undersampled multiframes," *IEEE Trans. Acoust., Speech, and Signal Processing*, pp. 1013–1027, June 1990.

[2] N. P. Galatsamos and R. T. Chin, "Digital restoration of multichannel images," *IEEE Trans. Acoust., Speech, and Signal Processing*, pp. 415–421, Mar. 1989.

[3] D. C. Ghiglia, "Space-invariant deblurring given independently blurred images of a common object," *J. Opt. Soc. Amer.* (A), vol. 1, Apr. 1984.

[4] B. R. Hunt and O. Kubler, "Karhunen-Loeve multispectral image restoration, Part I: Theory," *IEEE Trans. Acoust., Speech, and Signal Processing*, pp. 592–599, June 1984.

[5] A. Katsaggelos, "A multiple input image restoration approach," *J. Visual Comm. and Image Represent.*, vol. 1, no. 1, pp. 93–103, Sept. 1990.

[6] R. Bellman, *Introduction to Matrix Analysis*, 2nd ed. New York: McGraw-Hill, 1970.

[7] A. Papoulis, *Signal Analysis*. New York: McGraw-Hill, 1983.

[8] B. R. Hunt, "The application of constrained least squares estimation to image restoration by digital computer," *IEEE Trans. Computers*, vol. 22, pp. 805–812, Sept. 1973.

[9] R. C. Gonzalez and P. Wintz, *Digital Image Processing*, 2nd ed. Reading, MA: Addison-Wesley, 1987.

[10] A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-Posed Problems*. New York: Winston Wiley, 1977.

[11] L. L. Scharf, *Statistical Signal Processing*. Reading, MA: Addison-Wesley, 1990.

Time-Frequency Distribution Inversion of the Radon Transform

Berkman Sahiner and Andrew E. Yagle

Abstract— **In using filtered backprojection to compute the inverse Radon transform, the ramp filter amplifies noise. Spatially invariant noise filters reduce resolution. It is desirable to filter noise where projections have no local high-frequency components. Using the short-time Fourier transform, we apply a time-frequency mask filter that zeroes out projections where local signal energy is below a threshold. Results show improvement over reconstructions using spatially-invariant smoothing filters.**

I. INTRODUCTION

The Radon transform inversion problem is the basic problem of x-ray tomography. The problem is to reconstruct an image $\mu(x,y)$ from its projections $p(r,\theta)$ where

$$p(r,\theta) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mu(x,y)\delta(r - x\cos\theta - y\sin\theta)dxdy \qquad (1)$$

is the Radon transform of $\mu(x,y)$.

The most common procedure for image reconstruction from projections is filtered backprojection (FBP), in which the reconstruction is carried out in two stages. The first stage is filtering, in which the projections are filtered to yield the filtered projections $q(r,\theta)$ [1]. The second stage is backprojection, in which the image is obtained from filtered projections using

$$\mu(x,y) = \frac{1}{2\pi}\int_0^{\pi} q(x\cos\theta + y\sin\theta,\theta)d\theta. \qquad (2)$$

In the filtering stage, the frequency response of the ideal filter is $|w|$; this is called a ramp filter [2]. A problem with the ramp filter is that it amplifies the high-wavenumber or high-frequency components of both the noise and the data. Since noise usually dominates at high frequencies, it is common practice to use a low-pass filter $H(w)$ in conjunction with $|w|$ to improve the signal-to-noise ratio (SNR). Hence, in practice, we have

$$Q(w,\theta) = P(w,\theta)|w|H(w) \qquad (3)$$

where $Q(w,\theta)$ and $P(w,\theta)$ denote the Fourier transforms of $q(r,\theta)$ and $p(r,\theta)$ in the r variable, for each projection angle θ. Usually, $H(w)$ has some gentle rolloff characteristics at high frequencies to prevent ringing at edges. A typical shape for $|w|H(w)$ is shown in

Manuscript received April 22, 1992; revised February 16, 1993. This work was supported by the Office of Naval Research under Grant #N00014-90-J-1897. The associate editor coordinating the review of this paper and approving it for publication was Prof. Y. Bresler.

The authors are with the Department of Electrical Engineering and Computer Science, The University of Michigan, Ann Arbor, MI 48109.

IEEE Log Number 9210833.

1057–7149/93$03.00 © 1993 IEEE

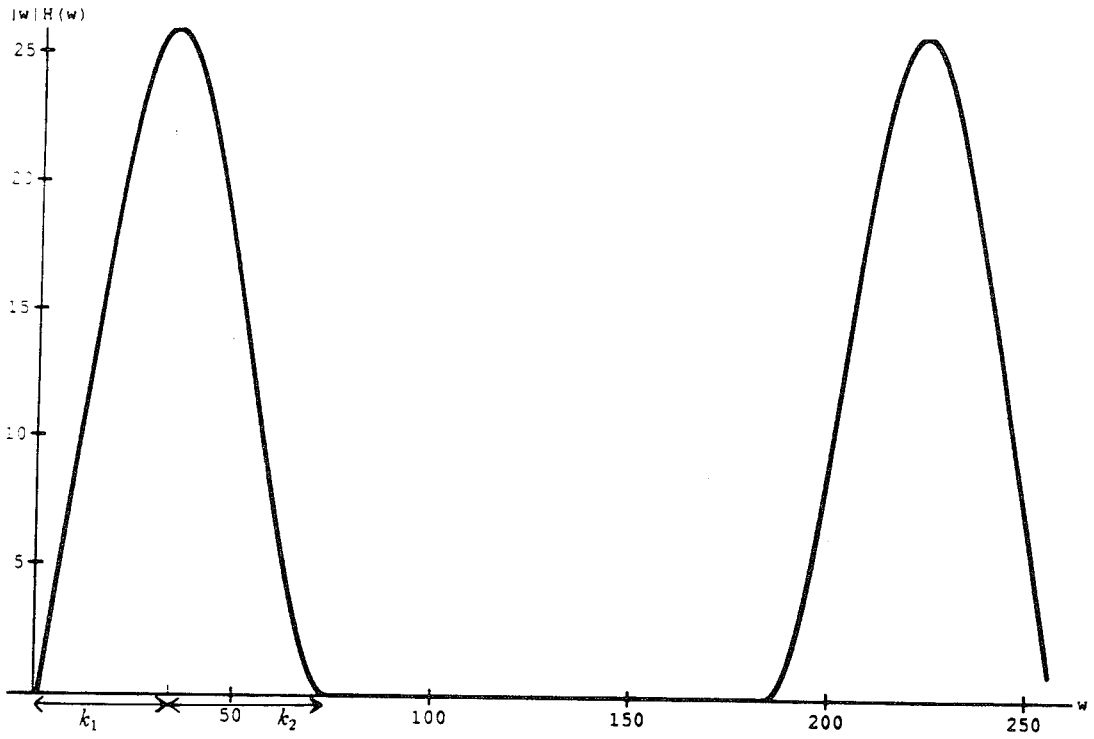


Fig. 1. A typical shape for $|w|H(w)$.

Fig. 1. The SNR improvement obtained by using $H(w)$ comes at the expense of degraded image resolution, since $H(w)$ will also smooth the edges in the image.

It is desirable to reduce the noise energy where the projection energy is small, while preserving the projection energy (and necessarily the noise energy) where it is large. However, spatially invariant filters are unable to have their characteristics vary in a spatially-varying manner, so this selectivity cannot be attained.

For this paper, we use time-frequency (t-f) distributions of $p(r,\theta)$ (considering r as the time variable and θ as a constant for each projection angle) to accomplish spatially-varying filtering. In the next section, we summarize the basic approach used in spatially-varying filtering based on a t-f distribution. In Section III, we concentrate on the short-time Fourier transform, which is chosen as the t-f distribution tool in this correspondence. We present the application to the inverse Radon transform and numerical results in Section IV.

II. Spatially-Varying Filtering Based on T-F Distributions

T-f distributions (representations) describe the intensity of a signal simultaneously in time and frequency. Such representations are of interest when dealing with nonstationary signals.

Numerous definitions of the t-f distribution $F(w,t)$ of a signal $f(t)$ have been suggested [3]. The most familiar representation is the short-time Fourier transform (STFT), which is reviewed in Section III. STFT is a member of a larger family called Cohen's class of distributions [3]; each member of the class has its advantages and drawbacks, and there is no single "correct" choice of a representation.

A signal $f(t)$ can be filtered in the time-frequency domain as follows. The first step is to multiply $F(w,t)$ by a filter $W(w,t)$ to yield [4]

$$G(w,t) = F(w,t)W(w,t). \tag{4}$$

The function $W(w,t)$ may be called a time-varying transfer function, since it represents the factor by which different frequency components of the local spectrum around the time t are multiplied. The multiplication is analogous to the classical time-invariant filtering in which $F(w)$ is multiplied by a filter transfer function $W(w)$.

If one is interested in recovering a signal from noise-corrupted data, then a typical choice for $W(w,t)$ is a one-zero mask [5]–[9]. $W(w,t)$ is set to unity in the region of the t-f plane where the energy of the signal is above a threshold [5], [8]. This region is called the region of support (ROS) of $f(t)$ in the t-f plane. If the signal energy is below the threshold, then $W(w,t)$ is set to zero.

After $G(w,t)$ is computed, the second step is computation of the function $g(t)$ whose t-f distribution is $G(w,t)$. $g(t)$ is then the time-frequency variant filtered $f(t)$. However, $G(w,t)$ may not be a valid t-f distribution, i.e., there may be no function $g(t)$ whose t-f distribution is $G(w,t)$. In this case, one possible solution is to find the signal $\hat{g}(t)$ whose t-f distribution best approximates $G(w,t)$ in the mean squared error (MSE) sense. Note that this error measure does not necessarily lead to the 'best' filtered signal, but is chosen because it is easy to deal with. Letting $\hat{G}(w,t)$ represent the t-f distribution of $\hat{g}(t)$, one finds $\hat{g}(t)$ such that

$$D = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} |\hat{G}(w,t) - G(w,t)|^2 dt dw \tag{5}$$

is minimized. We show how to find this $\hat{g}(t)$ for the STFT in the next section.

III. The Short-Time Fourier Transform

The STFT of a signal $f(t)$ is defined as [4]

$$F(w,t) = \int_{-\infty}^{\infty} f(\tau)h(t-\tau)e^{-jw\tau}d\tau \tag{6}$$

where $h(t)$ is called the analysis filter. The shape and length of $h(t)$ are important: Too short a filter may result in poor frequency resolution, and too long a filter may result in poor temporal resolution.

For a discrete-time signal $f(k)$, the discrete short-time Fourier transform (DSTFT) is defined as [10]:

$$F_{\Omega_M, R}(m,k) = \sum_{l=-\infty}^{\infty} f(l)h(Rk-l)e^{-j\Omega_M lm} \quad (7)$$

where M is the transform size, $\Omega_M = 2\pi/M$ is the sampling period in the frequency domain, R is the sampling period in the time domain, $m = 0, 1, \cdots, M-1$ is the frequency variable, and k is the time variable.

With mild conditions on the analysis filter (e.g., if $R < M$, the first R samples of $h(n)$ should be nonzero, or if R divides M, none of the R polyphases of $h(n)$ should be identically zero) $f(k)$ can be recovered exactly from $F(m,k)$ [11]. The inversion relation is

$$f(l) = \sum_{k=-\infty}^{\infty} s(l-kR)\frac{1}{M}\sum_{m=0}^{M-1} F_{\Omega_M, R}(m,k)e^{j\Omega_M ml} \quad (8)$$

where $s(l)$ is called the synthesis filter and is determined by the analysis filter $h(l)$. An algebraic approach to determine $s(l)$ for exact reconstruction is given in [12].

When we wish to perform time-frequency filtering to obtain a filtered signal $\hat{g}(k)$ from $f(k)$, we first multiply $F(m,k)$ by a mask $W(m,k)$ (compare to (4))

$$G(m,k) = F(m,k)W(m,k) \quad (9)$$

and then we find the function $\hat{g}(k)$ whose DSTFT is closest to $G(m,k)$. The solution to the problem of finding $\hat{g}(k)$ from $G(m,k)$ has been addressed in [13]. In particular, when the analysis filter length N is not larger than the transform length M, then the synthesis equation (8) applied to $G(m,k)$ gives the $\hat{g}(k)$ minimizing the discrete counterpart to (5).

IV. Application to Filtered Backprojection

Now suppose the projection data are noisy. Then (3) becomes

$$Q(w,\theta) = P_n(w,\theta)|w|H(w) \quad (10)$$

where the subscript n in (10) indicates that the projections are corrupted by noise.

Our idea is to spatially vary the bandwidth of $H(w)$. Let us define

$$R(m,k,\theta) = W(m,k,\theta)P_n(m,k,\theta) \quad (11)$$

for each angle θ where

$$W(m,k,\theta) = \begin{cases} 1 & \text{in the region of support (ROS)} \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

and $P_n(m,k,\theta)$ denotes the DSTFT of $p_n(r,\theta)$ for a fixed θ. Here $p_n(r,\theta)$ has been discretized in r. Notice that the radial variable r plays the role of the time variable in the DSTFT, i.e., we are applying a time-frequency representation to process the data in spatially-varying fashion.

The ROS is determined from *a priori* knowledge about it and the DSTFT of $p(r,\theta)$ with respect to the variable r. One possibility is to estimate the ROS using the threshold test given in Section II, i.e., define the ROS to be the region in which the DSTFT amplitude exceeds some threshold. We then obtain $\hat{p}(r,\theta)$ as the signal such that the MSE between the DSTFT of $\hat{p}(r,\theta)$ and $R(m,k,\theta)$ is minimized. For a suitable choice of the analysis filter h, this can be done using the synthesis equation given in Section III.

Using this $\hat{p}$, we proceed with (3) to find the filtered projections.

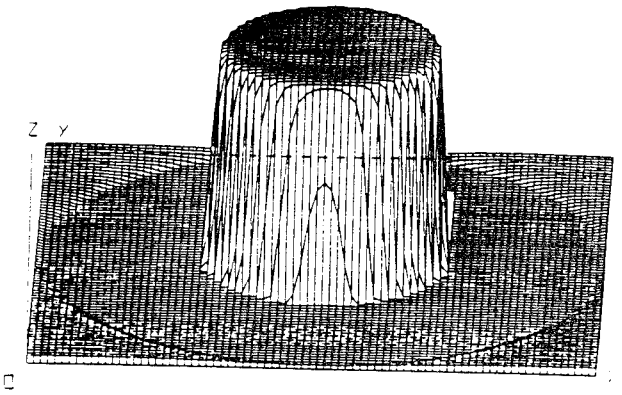


Fig. 2. Reconstruction from noiseless data.

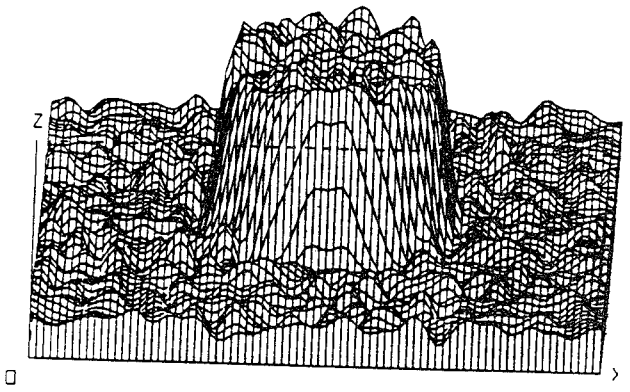


Fig. 3. The best reconstruction from noisy data using filtered backprojection and a spatially invariant filter.

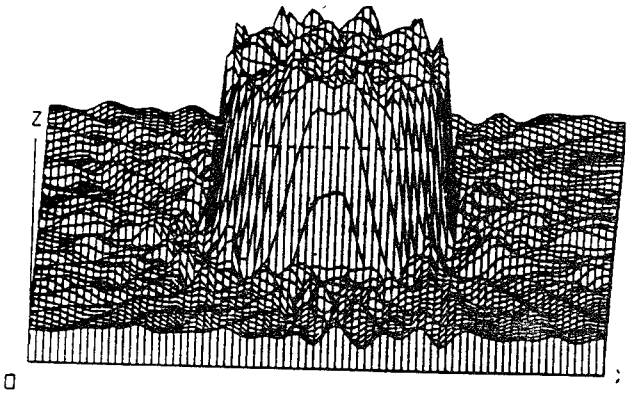


Fig. 4. The best reconstruction from noisy data using a t-f mask.

Example 1: Consider the following function, defined over the square $-1 \le x, y \le 1$:

$$\mu(x,y) = \begin{cases} 1 & x^2+y^2 \le R^2 \\ 0 & \text{otherwise} \end{cases}$$

$$\Longrightarrow p(r,\theta) = p(r) = \begin{cases} 2\sqrt{R^2-r^2} & r \le R \\ 0 & \text{otherwise.} \end{cases} \quad (13)$$

In the simulations below, we have used samples of $p(r,\theta)$ with 128 samples in the r variable (over $-1 \le r \le 1$) and 50 samples in the θ variable (over $0 \le \theta \le \pi$).

Reconstruction from noiseless $p(r,\theta)$ using FBP is given in Fig. 2. To carry out the backprojection, linear interpolation is used in the

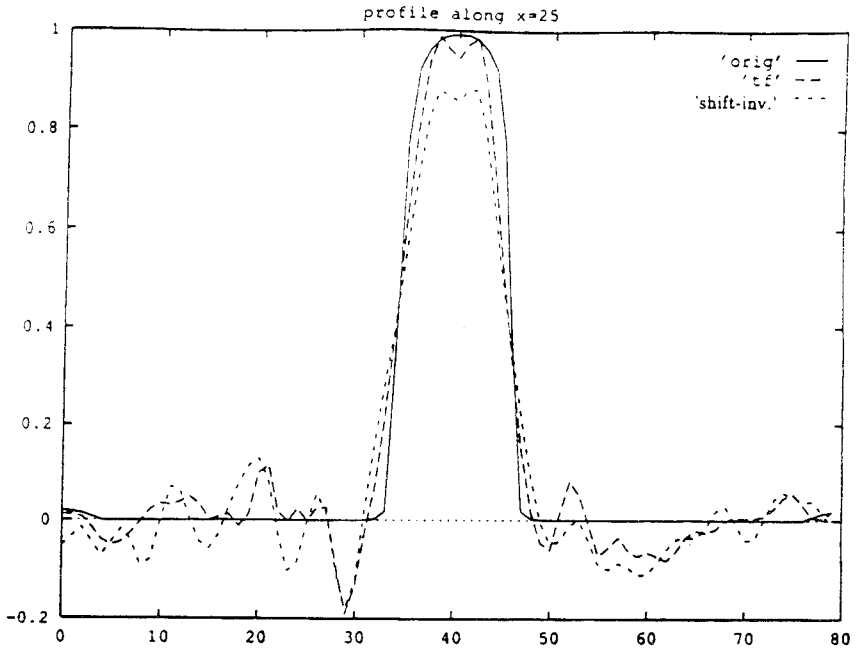


Fig. 5. Comparison of Figs. 2, 3, and 4 along the cross-section $x = 25$. Lines correspond to Fig. 2, short dashed lines correspond to Fig. 3, and long dashed lines correspond to Fig. 4.

radial variable, and a "small artifact" interpolation is employed in the angular variable since the number of angular samples is less than the number of radial samples [2].

Next, we reconstruct $\mu(x, y)$ from noisy projection data $p_n(r, \theta)$ obtained by adding white Gaussian noise with zero mean and variance 0.0036 to the projections. The "best" (in the sense of subjective human observation) reconstruction using FBP and a time-invariant filter $H(w)$ is given in Fig. 3. The optimal $H(w)$ was found by changing parameters k_1 and k_2 in Fig. 1 and choosing $k_1 = 21$ and $k_2 = 35$, which gave the best reconstruction.

We now present the result using t-f filtering. We choose our analysis filter to be Gaussian, $h(n) = e^{-n^2/16}$, with length 32 (i.e., we assume $h(k) = 0$ for $k \geq 16$ and $k < -16$). We also define the t-f mask $W(m, k)$ as

$$W(m, k) = \begin{cases} 1 & \text{if } |P(m, k)|^2 \geq \nu \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

where ν is a fixed threshold, found by choosing the value that gives the best reconstruction. The resulting reconstructed image is given in Fig. 4.

To compare the reconstructions of Fig. 3 and Fig. 4 with the noiseless reconstruction Fig. 2, we also plot a cross-section through each image along $x = 25$ in Fig. 5. The noiseless reconstruction is close to unity in the center of the image, as expected. Reconstruction with a t-f mask follows the noiseless reconstruction closely at the edges, due to the fact that the projection energy in the t-f plane is preserved where it is expected to be large. However, this does not result in increased noise energy (compared to space-invariant filtering) at other points in the image.

Example 2: We now apply the algorithms given in Example 1 to a frequently used phantom in medical imaging. The phantom is supposed to be a section of the human head with the denser (high μ) areas indicating tumors and the less dense areas indicating spinal fluid. The reconstruction from noiseless data is given in Fig. 6. The effect of the skull has been removed using bone deleting techniques [14].

Fig. 6. Reconstruction from noiseless data.

The additive noise is again Gaussian with variance 10^{-6}. The best reconstruction from noisy data using FBP and a time invariant filter is given in Fig. 7 ($k_1 = 34$ and $k_2 = 37$).

Reconstruction using a t-f mask is given in Fig. 8. Note that compared to Fig. 7, all of the larger, elliptical objects in Fig. 8 contain less noise; however, the same degree of resolution is maintained at the edges. Also, the noise in the background (which is ideally flat) is reduced, and therefore the small peaks in the front are more easily distinguished from the background.

V. CONCLUSION

We have applied the idea of time-frequency masking to the inversion of the Radon transform. This results in a spatially-varying filter which regularizes the $|w|$ filter, reduces the noise, and still preserves local high frequency features such as edges. Two examples illustrate the improvement over spatially- invariant filtering.

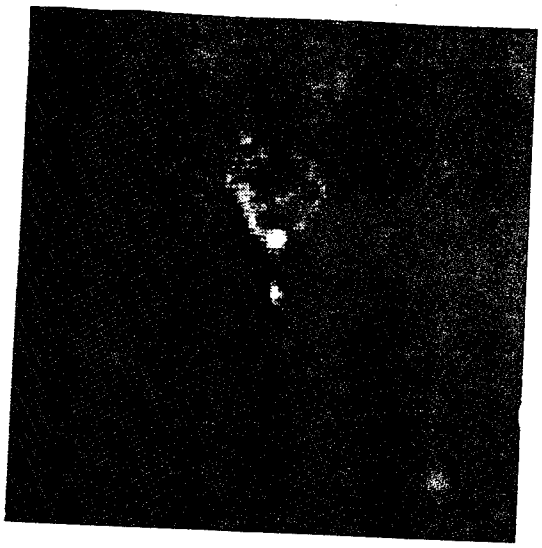


Fig. 7. The best reconstruction from noisy data using filtered backprojection and a spatially invariant filter.

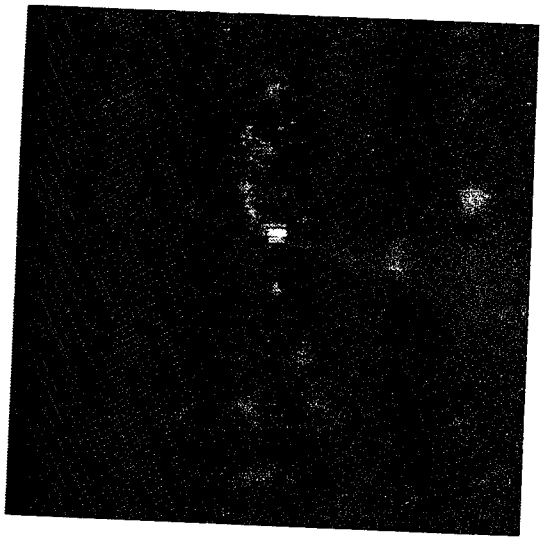


Fig. 8. The best reconstruction from noisy data using a t-f mask.

REFERENCES

[1] S. W. Rowland, "Computer Implementation of Image Reconstruction Formulas," in *Image Reconstruction from Projections: Implementation and Applications*, G. T. Herman, ed. (vol. 32 of *Topics in Applied Physics*). New York: Springer-Verlag, 1979.

[2] R. A. Brooks, G. H. Weiss, and A. J. Talbert, "A new approach to interpolation in computed tomography," *J. Compt. Assist. Tomog.*, vol. 2, pp. 577–585, 1978.

[3] L. Cohen, "Time-frequency distributions—a review," *Proc. IEEE*, vol. 77, pp. 941–981, 1989.

[4] B. E. A. Saleh and N. S. Subotic, "Time-variant filtering of signals in the mixed time-frequency domain," *IEEE Trans. Acoust., Speech, Sig. Proc.*, vol. 33, pp. 1479–1487, 1985.

[5] T. E. Koczwara and D. L. Jones, "On mask selection for time-varying filtering using the Wigner distribution," in *Proc. ICASSP-90*, pp. 2487-2490.

[6] J. Jeong and W. J. Williams, "Time-varying filtering and signal synthesis," in *Time-Frequency Signal Analysis*, B. Boashash, ed. Melbourne: Longman and Cheshire, 1991.

[7] B. Boashash and L. B. White, "Instantaneous frequency estimation and automatic time-varying filtering," in *Proc. ICASSP-1990*, pp. 1221–1224.

[8] M. Bikdash and K. B. Yu, "Linear shift varying filtering of nonstationary chirp signals," in *Proc. IEEE Southeastern Symposium on System Theory, 1988*, pp. 428–432.

[9] G. F. Boudreaux-Bartels and T. W. Parks, "Time-varying filtering and signal estimation using Wigner distribution synthesis techniques," *IEEE Trans. Acoust., Speech, Sig. Proc.*, vol. 34, pp. 442–451, 1988.

[10] M. R. Portnoff, "Representation of digital signals and systems based on the short-time Fourier transform," *IEEE Trans. Acoust., Speech, Sig. Proc.*, vol. 28, pp. 55–69, 1980.

[11] Z. Shpiro and D. Malah, "An algebraic approach to discrete short-time Fourier transform analysis and synthesis," in *Proc. ICASSP-84*, pp. 2.3.1–2.3.4, pp. 804–807.

[12] Z. Shpiro and D. Malah, "Design of filters for discrete short-time Fourier transform synthesis," in *Proc. ICASSP-85*, pp. 14.6.1–14.6.4, pp. 537–540.

[13] A. Dembo and D. Malah, "Signal synthesis from modified discrete short-time Fourier transform," *IEEE Trans. Acoust., Speech, Sig. Proc.*, vol. 34, pp. 168–180, 1988.

[14] G. Henrich, N. Mai, and M. Backmund, "Preprocessing in CT picture analysis: A bone deleting algorithm," *J. Compt. Assist. Tomog.*, vol 3, pp 379- 384, 1979.

A Derivative-Free Noncircular Fan-Beam Reconstruction Formula

Ge Wang, Tein-Hsiang Lin, and Ping-chin Cheng

Abstract—**In order to perform fan-beam reconstruction using projection data collected from a noncircular scanning locus, existing noncircular fan-beam formulas require a derivative of the scanning locus with respect to the rotation angle. In this paper, a derivative-free noncircular fan-beam reconstruction formula is obtained based on a geometrical explanation of the circular equispatial fan-beam reconstruction formula. A mathematical proof is then provided under the conditions that the source-to-origin distance is symmetric with respect to the origin of the reconstruction coordinate system, is differentiable almost everywhere and changes not too fast with respect to the rotation angle. The derivative-free noncircular fan-beam reconstruction formula is the same as the circular one, except that the source-to-origin distance is a function of the rotation angle. A typical simulation result of the noncircular fan-beam formula is given.**

I. INTRODUCTION

Fan-beam reconstruction was first studied in the case of a circular scanning locus [1], [2], and then extended into the case of

Manuscript received June 18, 1992; revised December 10, 1992. This work was supported by grants from the U. S. National Institute of Health—Biomedical Research Grant Program (BRSG SO7 RR07 066), the U. S. National Science Foundation (NSF MIP-9 010 948)), the U.S. Department of Energy (DE-AS08-88DP10 782, DE-FG03-89SF18 012), the Whitaker Foundation (Program for Biomedical Engineering); and by the NSERC (Canada) (Operating and Strategic grants), the Ontario Center for Materials Research, and the Canadian Synchrotron Radiation Facility (CSRF). The associate editor coordinating the review of this paper and approving it for publication was Dr. Y. Bresler.

G. Wang is currently with the Division of Radiology Research, Mallinckrodt Institute of Radiology, Washington University, St. Louis, MO 63110.

T. H. Lin and P. C. Cheng are with the Advanced Real Time System Laboratory (ARTS) and Advanced Microscopy and Imaging Laboratory (AMIL), Department of Electrical and Computer Engineering, Center for Electronic and Electrooptics Materials (CEEM), State University of New York, Buffalo, NY 14260.

IEEE Log Number 9210839.

1057–7149/93$03.00 © 1993 IEEE

APPENDIX K2

B. Sahiner and A.E. Yagle, "Time-Frequency Distribution Inversion of the Radon Transform," IEEE 1991 Medical Imaging Conference, Santa Fe, NM, Nov. 5-9, 1991, pp. 2043-2047.

This is the conference paper version of Appendix K1.

TIME-FREQUENCY DISTRIBUTION INVERSION OF THE RADON TRANSFORM

Berkman Sahiner and Andrew E. Yagle
Dept. of Electrical Engineering and Computer Science
The University of Michigan, Ann Arbor, Michigan 48109

Abstract

In using filtered backprojection to compute the inverse Radon transform, the Hilbert transform-derivative filtering operation amplifies noise. Filtering the noise generally reduces resolution and smoothes edges. It would be desirable to spatially filter the noise where the projections do not have high-frequency components, and accept the noise where high-frequency components are present, since these must be preserved for the Hilbert transform-derivative. Using the short-time Fourier transform, we apply a time-frequency mask filter that zeroes out each projection in locations where local signal energy is below a threshold. Results show improvement over reconstructions using time-invariant smoothing filters.

1 Introduction

The Radon transform inversion problem is the basic problem of x-ray tomography. The problem is to reconstruct an image $\mu(x,y)$ from its projections $p(r,\theta)$ where

$$p(r,\theta) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mu(x,y)\delta(r - x\cos\theta - y\sin\theta)dxdy \quad (1)$$

is the Radon transform of $\mu(x,y)$

The most common procedure for inverting the Radon transform is filtered backprojection (FBP), in which the projections $p(r,\theta)$ are first filtered to yield

$$Q(w,\theta) = P(w,\theta)|w| \quad (2)$$

where $P(w,\theta)$ is the Fourier transform of $p(r,\theta)$ and $|w|$ performs the Hilbert transform and derivative operations. The image $\mu(x,y)$ is obtained by backprojecting $q(r,\theta)$, the inverse Fourier transform of $Q(w,\theta)$ using

$$\mu(x,y) = \frac{1}{2\pi}\int_0^{\pi} q(x\cos\theta + y\sin\theta,\theta)d\theta \quad (3)$$

A problem with FBP is that the filter $|w|$ amplifies the high-wavenumber or high-frequency components of both the noise and the data. Since noise usually dominates at high frequencies, it is common practice to use a low-pass filter $H(w)$ in conjuction with $|w|$ to improve the signal-to-noise ratio (SNR). Hence, in practice, we have

$$Q(w,\theta) = P(w,\theta)|w|H(w) \quad (4)$$

where $H(w)$ has some gentle rolloff characteristics at high frequencies to prevent ringing at edges. A typical shape for $|w|H(w)$ is shown in Figure 1. However, the SNR improvement obtained by using $H(w)$ comes at the expense of degraded image resolution, since $H(w)$ will also smooth the edges in the image.

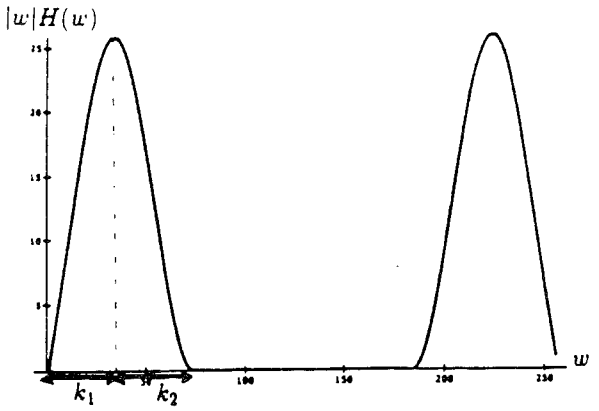


Figure 1: A typical shape for $|w|H(w)$

It is desirable to reduce the noise energy where the projection energy is small, while preserving the projection energy (and necessarily the noise energy) where it is large. But spatially invariant filters are unable to have their characteristics vary in a nonstationary manner, so this selectivity cannot be attained.

In this paper, we use time-frequency (t-f) distributions of $p(r,\theta)$ (considering r as the time variable and θ as a constant for each projection angle) to accomplish spatially variant filtering. In the next section, we summarize the basic approach used in spatially varying filtering based on a t-f distribution. In Section III, we concentrate on the short-time Fourier transform, which is chosen as the t-f distribution tool in this paper. We present the application to Inverse Radon transform and numerical results in Section IV.

0-7803-0513-2/92$0.3.00 ©IEEE

2 Spatially-Varying Filtering Based on T-F Distributions

T-f distributions (representations) describe the intensity of a signal simultaneously in time and frequency. Such representations are of interest when one is dealing with nonstationary signals.

Numerous definitions of the t-f distribution $F(w,t)$ of a signal f(t) have been suggested [1, 2, 3]. The most familiar representation is the short-time Fourier transform (STFT), which is reviewed in Section III. STFT is a member of a larger family, called Cohen's class of distributions [4]. Each member of the class has its advantages and drawbacks, and there is no single "correct" choice of a representation.

A signal $f(t)$ can be filtered in the time-frequency domain as follows. First, multiply $F(w,t)$ by a filter $W(w,t)$ [5], yielding

$$G(w,t) = F(w,t)W(w,t) \tag{5}$$

The function $W(w,t)$ may be called a time-varying transfer function, since it represents the factor by which different frequency components of the local spectrum around the time t are multiplied. The multiplication is analogous to the classical time-invariant filtering in which $F(w)$ is multiplied by a filter transfer function $W(w)$.

If one is interested in recovering a signal from noise-corrupted data, then a typical choice for $W(w,t)$ is a one-zero mask [6]. $W(w,t)$ is set to 1 in the region of the t-f plane where the energy of the signal is above a threshold. This region is called the Region of Support (ROS) of $f(t)$ in the t-f plane. If the signal energy is below the threshold, then $W(w,t)$ is set to 0. The threshold value is usually set arbitrarily, and little is understood about the effects of the size of the one-zero mask on the filtered signal [6].

Once $G(w,t)$is computed, one can proceed to find the function $g(t)$ whose t-f distribution is $G(w,t)$. $g(t)$ is then the spatially-variant filtered $f(t)$. However, $G(w,t)$ may not be a valid t-f distribution. That is, there may be no function $g(t)$ whose t-f distribution is $G(w,t)$. In this case, one tries to find the signal $\hat{g}(t)$ whose t-f distribution best approximates $G(w,t)$ in the mean squared error sense. Letting $\hat{G}(w,t)$ represent the t-f distribution of $\hat{g}(t)$, one finds $\hat{g}(t)$ such that

$$D = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} |\hat{G}(w,t) - G(w,t)|^2 dtdw \tag{6}$$

is minimized. We show how to find this $\hat{g}(t)$ for the STFT next.

3 The Short-Time Fourier Transform

The STFT of a signal f(t) is defined as [5]

$$F(w,t) = \int_{-\infty}^{\infty} f(\tau)h(t-\tau)e^{-jw\tau}d\tau \tag{7}$$

where $h(t)$ is called the analysis filter or the window function. The shape of $h(t)$ and its length L are important. Too short a window may result in poor frequency resolution and too long a window may result in poor temporal resolution.

For a discrete-time signal $f(k)$, the discete short-time Fourier transform (DSTFT) is defined as [7]:

$$F_{\Omega_M,R}(m,k) = \sum_{l=-\infty}^{\infty} f(l)h(Rk-l)e^{-j\Omega_M lm} \tag{8}$$

where

M is the transform size

$\Omega_M = 2\pi/M$ is the sampling period in the frequency domain

R is the sampling period in the time domain

$m = 0, 1, \cdots, M-1$ is the frequency variable

k is the time variable.

With a mild condition on the analysis filter, $f(k)$ can be recovered exactly from $F(m,k)$ for $R < M$ [8]. The inversion relation is:

$$f(l) = \sum_{k=-\infty}^{\infty} s(l-kR)\frac{1}{M}\sum_{m=0}^{M-1} F_{\Omega_M,R}(m,k)e^{j\Omega_M ml} \tag{9}$$

where $s(l)$ is called the synthesis filter and is dictated by the analysis filter $h(l)$. An algebraic approach to determine $s(l)$ for exact reconstruction is given in [9].

When we wish to do time-frequency filtering to obtain a filtered signal $\hat{g}(k)$ from $f(k)$, we first multiply $F(m,k)$ by a mask $W(m,k)$ (compare to (5))

$$G(m,k) = F(m,k)W(m,k) \tag{10}$$

and then we find the function $\hat{g}(k)$ whose DSTFT is closest to $G(m,k)$. The solution to the problem of finding $\hat{g}(k)$ from $G(m,k)$ has been addressed in [10]. In particular, when the analysis filter length N is not larger than the transform length M, then the synthesis equation (9) applied to $G(m,k)$ gives the $\hat{g}(k)$ minimizing the discrete counterpart to (6).

4 Numerical Results

Consider (4) again

$$Q(w,\theta) = P_n(w,\theta)|w|H(w) \tag{11}$$

where the subscript n in (11) indicates that the projections are corrupted by noise.

Our idea is to spatially vary the bandwidth of $H(w)$. Let us define

$$R(m,k,\theta) = W(m,k,\theta)P_n(m,k,\theta) \tag{12}$$

for each angle θ where

$$W(m,k,\theta) = \begin{cases} 1 & \text{in the region of support (ROS)} \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

and $P_n(m,k,\theta)$ denotes the DSTFT of $p_n(r,\theta)$ for a fixed θ. Here, $p_n(r,\theta)$ has been discretized in r.

The region of interest is determined from a priori knowledge about the ROS and the DSTFT of $p(r,\theta)$ with respect to the variable r. One possibilty for determining the region of interest is to estimate the ROS using the threshold test given in Section II. That is, define the ROS to be the region in which the signal amplitude exceeds some threshold.

We then obtain $\hat{p}(r,\theta)$ as the signal such that the MSE between the DSTFT of $\hat{p}(r,\theta)$ and $R(m,k,\theta)$ is minimized. For a suitable choice of the analysis filter h, this can be done using the synthesis equation given in Section III.

Using this $\hat{p}$, we proceed with (4) to find the filtered projections.

Example 1 : Consider the following function, defined over the square $-1 \le x \le 1,\ -1 \le y \le 1$

$$\mu(x,y) = \begin{cases} 1 & x^2+y^2 \le R^2 \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

$$\Longrightarrow p(r,\theta) = p(r) = \begin{cases} 2\sqrt{R^2-r^2} & r \le R \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

In the simulations below, we have used samples of $p(r,\theta)$ with 128 samples in the r variable (over $-1 \le r \le 1$) and 50 samples in the θ variable (over $0 \le \theta \le \pi$).

Reconstruction from noiseless $p(r,\theta)$ using FBP is given in Figure 2. To carry out the backprojection, linear interpolation is used in the r variable and a "small artifact" interpolation is employed in the angular variable [11].

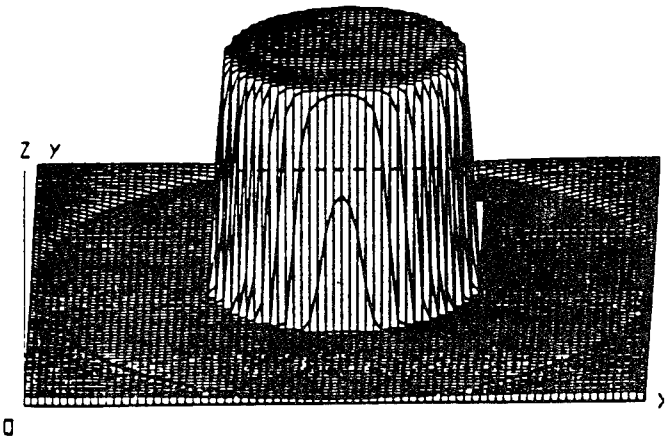


Figure 2: Reconstruction of $\mu(x,y)$ for Example 1, with noiseless data

Next, we reconstruct $\mu(x,y)$ from noisy projection data $p_n(r,\theta)$ obtained by adding white Gaussian noise with zero mean and variance 3.6×10^{-3} to the projections. The "best" (in the sense of subjective human observation) reconstruction using FBP and a time-invariant filter $H(w)$ is given in Figure 3. The optimal $H(w)$ was found by changing parameters k_1 and k_2 in Figure 1 and choosing the values that give the best reconstruction, which are $k_1 = 21$ and $k_2 = 35$.

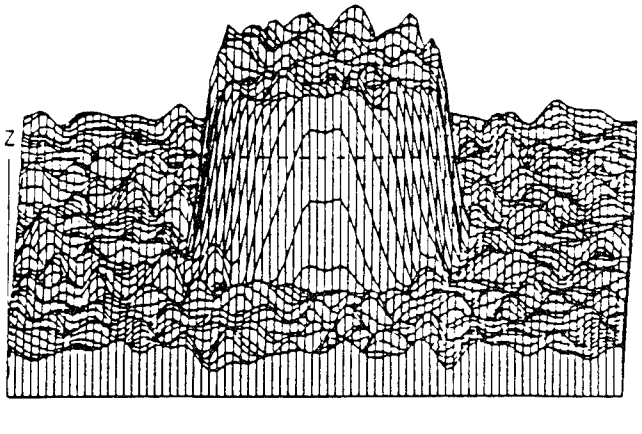


Figure 3: The best reconstruction from noisy data with a shift invariant filter

We now present the result using t-f filtering. In order to use the time-frequency approach, we first choose our analysis filter to be Gaussian,

$$h(n) = e^{-n^2/16}$$

with length 32 (i.e. we assume $h(k) = 0$ for $|k| \ge 16$). We also define the t-f mask $W(m,k)$ as

$$W(m,k) = \begin{cases} 1 & \text{if } |P(m,k)|^2 \ge \nu \\ 0 & \text{otherwise} \end{cases} \tag{16}$$

where ν is a fixed threshold, found by choosing the value that gives the best reconstruction.

The resulting reconstructed image is given in Figure 4.

To compare the reconstructions of Figure 3 and Figure 4 with the noiseless reconstruction, we also plot a cross-section through each image along $x = 25$ in Figure 5. The noiseless reconstruction is close to unity in the center of the image, as expected. Reconstruction with a t-f mask follows the noiseless reconstruction closely at the edges,

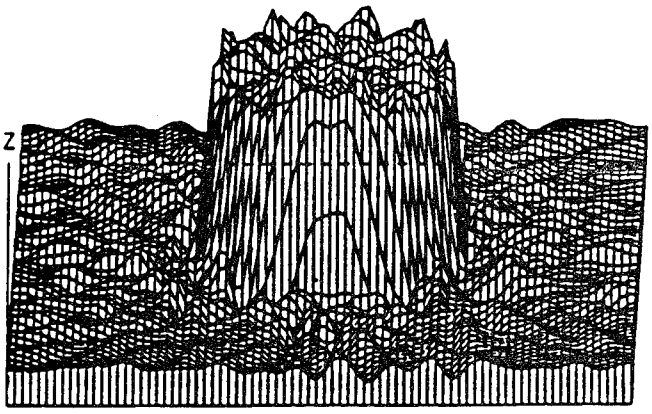


Figure 4: The best reconstruction from noisy data with a t-f mask

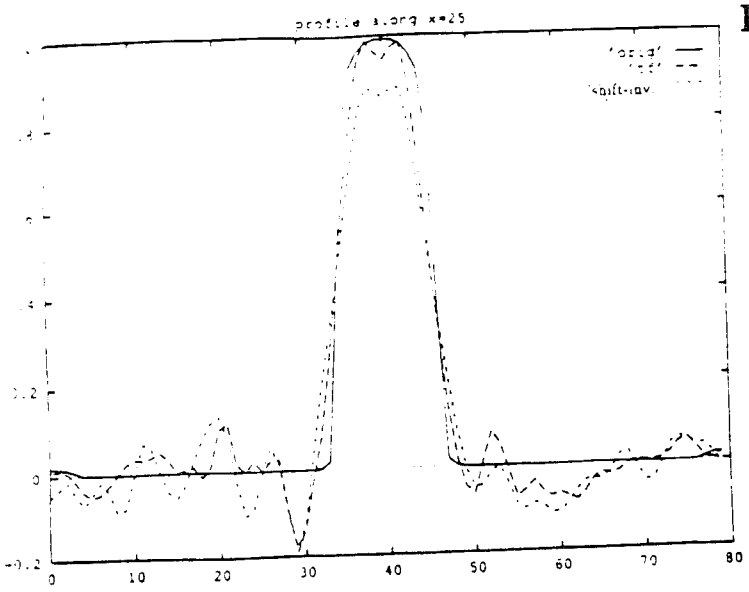


Figure 5: Comparison of Figures 2, 3, and 4 along the cross section x=25. Lines correspond to Figure 2, short dashed lines to Figure 3, and long dashed lines to Figure 4.

due to the fact that the projection energy in the t-f plane is preserved where it is expected to be large. However, this does not result in increased noise energy (compared to space-invariant filtering) at other points in the image.

Example 2 : We now apply the algorithms given in Example 1 to a frequently-used phantom in medical imaging. The phantom is supposed to be a section of the human head with the denser (high μ) areas indicating tumors and the less dense areas indicating spinal fluid. The reconstruction from noiseless data is given in Figure 6. The effect of the skull has been removed using bone deleting techniques [12].

The best reconstruction from noisy data (noise variance=10^{-4}) using FBP and a time invariant filter is given in Figure 7, found by choosing $k_1 = 34$ and $k_2 = 37$. Reconstruction using a t-f mask is given in Figure 8.

Note that in Figure 8, the oval, high-density area in the center contains less noise and hence is easier to notice. Also, the noise in the background (which is ideally flat) is reduced, and therefore the small peaks in front of the oval area are more easily distinguished from the background.

5 Conclusions

We have applied the idea of time-frequency masking to the inversion of the Radon transform. This results in a spatially-varying filter which regularizes the $|w|$ filter, reduces the noise, and still preserves local high frequency features such as edges. Two examples illustrate the improvement over spatially-invariant filtering.

References

[1] A. W. Rihaczek, "Signal energy distribution in time and frequency," IEEE Trans. Inform. Theory, vol IT-14, p. 369, 1968.

[2] T. A. C. M. Claasen and W. F. G. Mecklenbrauker, "The Wigner distribution - A tool for time-frequency signal analysis Part I: Continuous-time signals," Phillips J. Res., vol. 35, pp. 217-389, 1980.

[3] H. I. Choi and W. J. Williams, "Improved time-frequency representation of multicomponent signals using exponential kernels," IEEE Trans. Acoust., Speech, Sig. Proc., vol. 37, pp. 862-871, 1989.

[4] L. Cohen, " Time-frequency distributions - a review," Proc. IEEE, Vol 77, pp. 941-981, July 1989.

[5] B. E. A. Saleh and N. S. Subotic, "Time-variant filtering of signals in the mixed time-frequency domain," IEEE Trans. Acoust., Speech, Sig. Proc., vol. 33, pp.1479-1487, 1985.

[6] T. E. Koczwara and D.L. Jones, "On mask selection for time-varying filtering using the Wigner Distribution," in Proc. ICASSP 1990, pp. 2487-2490.

[7] M. R. Portnoff "Representation of digital signals and systems based on the Short Time Fourier Transform," IEEE Trans. Acoust., Speech, Sig. Proc., vol. 28, pp. 55-69, 1980.

[8] Z. Shpiro and D. Malah, "An algebraic approach to discrete short-time Fourier transform analysis and synthesis," IEEE ICASSP-84, pp. 2.3.1-2.3.4, pp. 804-807.

[9] Z. Shpiro and D. Malah, "Design of filters for Discrete Short Time Fourier Transform synthesis," IEEE ICASSP-85, pp. 14.6.1-14.6.4, pp. 537-540.

[10] A. Dembo and D. Malah, "Signal synthesis from modified Discrete Short Time Fourier Transform," IEEE Trans. Acoust., Speech, Sig. Proc., vol. 34, pp.168-180, Feb. 1988.

[11] R. A. Brooks, G. H. Weiss and A. J. Talbert, "A new approach to interpolation in computed tomography," J. Comput. Assist. Tomog., vol 2 pp. 577-585, 1978.

[12] G. Henrich, N. Mai and M. Backmund, "Preprocessing in CT picture analysis: A bone deleting algorithm," J. Comput. Assist. Tomogr., Vol 3, pp 379-384, 1979.

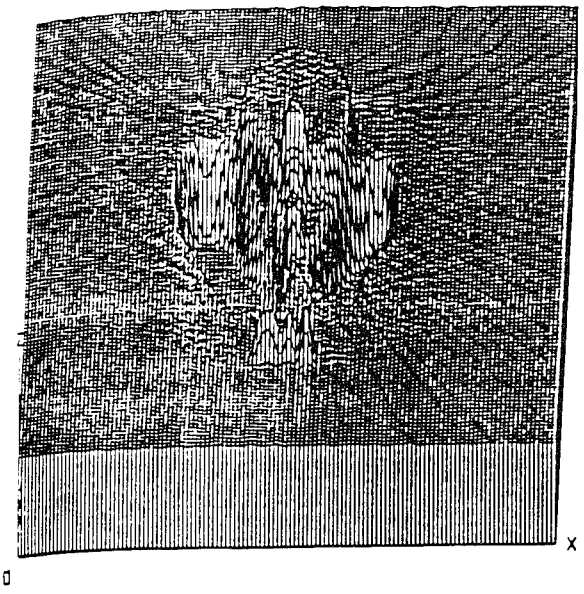


Figure 6: Reconstruction with noiseless data for Example 2

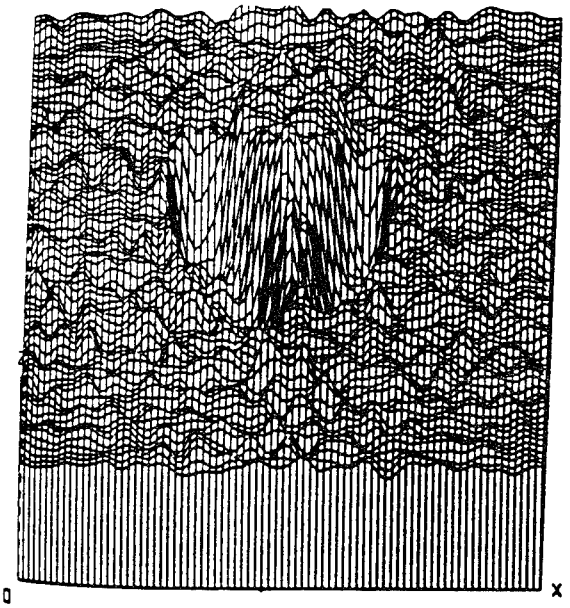


Figure 7: Reconstruction from noisy data with a shift invariant filter

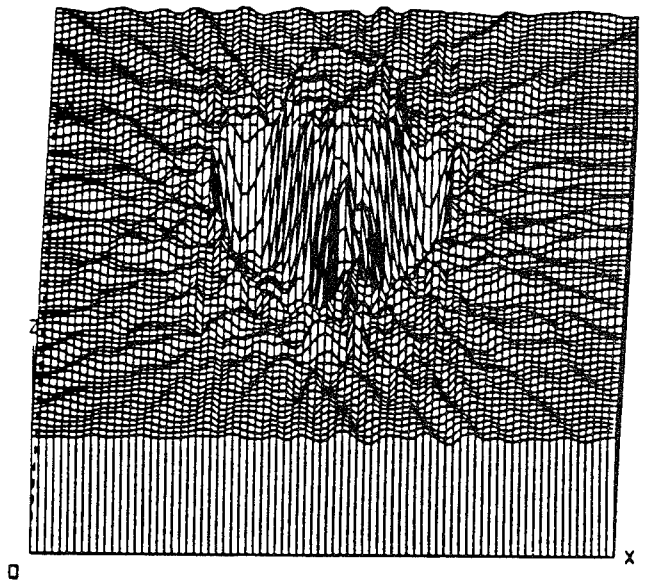


Figure 8: Reconstruction from noisy data with a t-f mask

APPENDIX L1

B. Sahiner and A.E. Yagle, "A Fast Algorithm for Backprojection with Linear Interpolation," to appear in IEEE Trans. Image Proc. 2(4), October 1993.

This paper derives a simple fast algorithm for backprojection in the inverse Radon transform. Interpolating and backprojecting four views at once saves half the multiplications.

Fig. 3(a) shows reconstructed Shepp and Logan's head phantom image using the circular fan-beam formula with a scanning circle of diameter 6. Fig. 3(b) shows the reconstructed image corresponding to Fig. 2(a) using our noncircular fan-beam formula with a scanning square of side-length 6. Fig. 3(c) shows the plots of the $y = 0.198$ line for Fig. 3(a) and 3(b). It can be observed that the reconstructed results obtained using different fan-beam formulas are almost the same in the reconstruction region. More simulation results were presented in [10]. The parameters of the above simulation are typical in our x-ray microtomographic system [11]. It can be verified that the square scanning locus used indeed meets all three conditions.

V. Conclusion

As far as the mathematical form is concerned, the proposed formula is the simplest among noncircular fan-beam formulas, as it is the same as the circular fan-beam formula [9] except that the source-to-origin distance depends on the rotation angle. However, we would like to emphasize that our formula is exact only with a symmetric scanning locus. If this symmetry condition is violated, other noncircular fan-beam formulas should be used for exact reconstruction, provided that the relevant conditions required are satisfied.

The main difference between Weinstein's formula [3] or Gullberg's formula [5] and the proposed formula lies in that the new formula requires no derivative of the scanning locus with respect to the rotation angle. Smith's extended fan-beam formula also contains a derivative of the scanning locus [4]. For an irregular scanning locus, it may not be trivial to estimate accurately its derivative. For example, in our x-ray microtomographic study, the scanning locus is subject to random interferences introduced by the mechanical rotation of the specimen stage [11]–[13]. As a result, a precise estimation of the derivative of the scanning locus is particularly difficult. On the other hand, the practical scanning locus used in x-ray microtomography can be made to meet our three conditions [11]–[13]. Without using the derivative of the scanning locus, the proposed formula will not be affected by the error in estimating the derivative.

Acknowledgment

The authors are grateful to reviewers for their valuable comments. The photos in this paper were prepared at the Division of Radiology Research, Mallinckrodt Institute of Radiology, Washington University, with the technical assistance of Professor M. W. Vannier, Mr. B. Brunsden, Mr. N. Hente, Mr. R. Knapp, and Mr. R. Walkup.

References

[1] G. T. Herman and A. Naparstek, "Fast image reconstruction based on a Radon inversion formula appropriate for rapidly collected data," *SIAM J. Appl. Math.*, vol. 33, pp. 511–533, Nov. 1977.

[2] B. K. P. Horn, "Fan-beam reconstruction methods," in *Proc. IEEE*, vol. 67, pp. 1616–1623, 1979.

[3] F. S. Weinstein, "Formation of images using fan beam scanning and noncircular source motion," *J. Opt. Soc. Amer.*, vol. 70, pp. 931–935, 1980.

[4] B. D. Smith, "Derivation of the extended fan-beam formula," *IEEE Trans. Med. Imag.*, vol. MI-4, pp. 177–184, 1985.

[5] G. T. Gullberg and G. L. Zeng, "A cone-beam filtered backprojection reconstruction algorithm for cardiac single photon emission computed tomography," *IEEE Trans. Med. Imag.*, vol. 11, no. 1, pp. 91–101, 1992.

[6] H. Hu, G. T. Gullberg, and R. A. Kruger, "Convolutional reconstruction algorithm for fan beam concave and convex circular detectors," *IEEE Trans. Med. Imag.*, vol. 7, no. 3, pp. 233–238, 1988.

[7] G. T. Gullberg, C. R. Crawford, and B. M. W. Tsui, "Reconstruction algorithm for fan beam with a displaced center-of-rotation," *IEEE Trans. Med. Imag.*, vol. 5, no. 1, pp. 23–29, 1986.

[8] C. R. Crawford and G. T. Gullberg, "Reconstruction for fan beam with an angular-dependent displaced center-of-rotation," *Med. Phys.*, vol. 15, no. 1, pp. 67–71, 1988.

[9] A. C. Kak and M. Slaney, *Principles of Computerized Tomographic Imaging*. Piscataway, NJ: IEEE Press, 1987.

[10] G. Wang, T. H. Lin, and P. C. Cheng, "A new extended fan-beam reconstruction formula," in *Proc. SPIE, vol. 1660*, pp. 262–273, 1992.

[11] G. Wang, T. H. Lin, P. C. Cheng, and D. M. Shinozaki, "A general cone-beam reconstruction algorithm," to appear in the Sept. 1993 issue of *IEEE Trans. on Med. Imag.*

[12] G. Wang, T. H. Lin, P. C. Cheng, and D. M. Shinozaki, "Preliminary error analysis on the general cone-beam reconstruction algorithm," in *Proc. SPIE, vol. 1660*, pp. 274–285, 1992.

[13] T. H. Lin, G. Wang, and P. C. Cheng, "Error analysis of the general cone-beam reconstruction algorithm for X-ray microtomography," *J. Scanning Microscopy, Suppl. II (Addendum)*, pp. 20–23, 1992.

A Fast Algorithm for Backprojection with Linear Interpolation

Berkman Sahiner and Andrew E. Yagle

***Abstract*— In the filtered backprojection procedure for image reconstruction from projections, backprojection dominates the computation time. We propose a simple algorithm that reduces the number of multiplications in linear interpolation and backprojection stage by 50%, with a small increase in the number of additions. The algorithm performs the interpolation and backprojection of four views together. Examples of implementation are given and extension to interpolation of more than four views is discussed.**

I. Introduction

The basic problem of x-ray tomography is to reconstruct an image $\mu(x, y)$ from its projections $p(r, \theta)$ where

$$p(r,\theta) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mu(x,y)\delta(r - x\cos\theta - y\sin\theta)dxdy \quad (1)$$

is the Radon transform of $\mu(x, y)$.

The most common procedure for image reconstruction from projections is filtered backprojection (FBP), in which the reconstruction is carried out in two stages. The first stage is filtering, in which the projections are filtered by a modified ramp filter to yield the filtered projections $q(r, \theta)$ [1]. The second stage is backprojection, in which the desired image $\mu(x, y)$ is obtained from filtered projections by backprojection.

In practical problems, we have only samples of $p(r, \theta)$. Let us denote

$$P_i(R) = p(r,\theta_i)\Big|_{r=Rd,\theta_i=\frac{\pi}{N}(i-1)}, R = 0,\cdots,M-1, i = 1,\cdots,N \quad (2)$$

Manuscript received June 12, 1992; revised February 5, 1993. This work was supported by the Office of Naval Research under Grant N00014-90-J-1897. The associate editor coordinating the review of this paper and approving it for publication was Prof. M. I. Sezan.

The authors are with the Department of Electrical Engineering and Computer Science, The University of Michigan, Ann Arbor, MI 48109.

IEEE Log Number 9210833.

1057–7149/93$03.00 © 1993 IEEE

where N is the total number of views, M is the number of samples in each view, and d is the radial sampling distance.

Similarly, let $Q_i(R)$ denote the discrete filtered projections, obtained by filtering $P_i(R)$ using a modified ramp filter. Then the backprojection operation is

$$\mu(x,y) = \frac{\pi}{N}\sum_{i=1}^{N} q_i(x\cos\theta_i + y\sin\theta_i) \tag{3}$$

where $q_i(r) = q(r,\theta_i)$ is obtained from $Q_i(R)$ by some kind of interpolation.

In the FBP method, backprojection dominates the computation time. The filtering stage can be done using the fast Fourier transform (FFT), which requires $O(NM\log_2 M)$ operations. On the other hand, in the backprojection stage, interpolation needs to be done for every x, y, and i in (3); therefore if linear interpolation (requiring 1 multiplication per interpolated point) is used, NL^2 multiplications and $3NL^2$ additions are needed to compute an $L \times L$ image.

In this paper, we propose a modification in the linear interpolation-backprojection stage which reduces the number of multiplications by about 50%, with a slight increase in the number of additions. This may result in a significant savings in computational time for processors which perform multiplications slower than additions. It will also result in reduced chip area in a VLSI implementation, since multipliers consume more chip area than adders.

There are a number of fast backprojection techniques, e.g. [2], [3], and [4], which reduce the number of operations even more than our technique. In the next section, we first present the basic idea and then compare our technique with other existing fast algorithms. In Section III, we discuss some implementation issues and give examples. A generalization of the basic idea is presented in Section IV.

II. The Fast Algorithm

If linear interpolation is used in (3), then $q_i(r)$ is approximated as

$$q_i(r) = Q_i(R) + Fr(r)\Delta Q_i(R) \tag{4}$$

where $R = [r]$ is the largest integer less than r, $Fr(r) = r - R$ is the fractional part of r, and the Δ operator acting on a discrete function is defined as $\Delta f(k) = f(k+1) - f(k)$.

We now show that if we backproject *four* views, namely the views numbered i, $i + N/4$, $i + N/2$, and $i + 3N/4$, then multiplications by $Fr(r)$ above can be combined.

For notational convenience, let us denote

$$\bar{q}_{i,j}(x,y) = q_{i-\frac{N}{4}j}\left(r = x\cos(\theta_i + \frac{\pi}{4}j) + y\sin(\theta_i + \frac{\pi}{4}j)\right)$$

$$j = 0,1,2,3 \tag{5}$$

and rewrite (3) as

$$\mu(x,y) = \frac{\pi}{N}\sum_{i=1}^{N/4}\sum_{j=0}^{3}\bar{q}_{i,j}(x,y). \tag{6}$$

Let us explicitly write $\bar{q}_{i,0}(x,y)$ and $\bar{q}_{i,2}(x,y)$ in terms of filtered projections $Q_i(R)$ as

$$\bar{q}_{i,0}(x,y) = Fr(r_0)\Delta Q_i(R_0) + Q_i(R_0) \tag{7}$$

where $R_0 = [r_0]$ and $r_0 = x\cos\theta_i + y\sin\theta_i$. Similarly,

$$\bar{q}_{i,2}(x,y) = Fr(r_2)\Delta Q_{i+N/2}(R_2) + Q_{i+N/2}(R_2) \tag{8}$$

where $r_2 = -x\sin\theta_i + y\cos\theta_i$. We find that to compute $\bar{q}_{i,0}(x,y)$ and $\bar{q}_{i,2}(x,y)$, we need to perform two multiplications, one by $Fr(r_0)$ and another by $Fr(r_2)$.

Let us now express $\bar{q}_{i,1}(x,y)$ and $\bar{q}_{i,3}(x,y)$ using the same factors. Applying the sine and cosine addition formulas to (5) yields

$$\bar{q}_{i,1}(x,y) = q_{i+N/4}\left(r = \frac{\sqrt{2}}{2}(r_0 + r_2)\right). \tag{9}$$

If we have samples of $q_{i+N/4}$ with a sampling interval of $\sqrt{2}/2$, i.e., if we have the discrete signal

$$Q'_i(R) = q_i(r)|_{r=R/\sqrt{2}},\ R = 0,1,2,\cdots,[M\sqrt{2}] \tag{10}$$

then $\bar{q}_{i,1}(x,y)$ can be expressed as

$$\bar{q}_{i,1}(x,y) = Fr(r_1)\Delta Q'_{i+N/4}(R_1) + Q'_{i+N/4}(R_1) \tag{11}$$

where $r_1 = r_0 + r_2$ and $R_1 = [r_1]$. However, using the fact that

$$Fr(r_1) = Fr(r_0 + r_2) = Fr(Fr(r_0) + Fr(r_2)) \tag{12}$$

equation (11) can be written as

$$\begin{aligned}\bar{q}_{i,1}(x,y) =& Fr(r_2)\Delta Q'_{i+N/4}(R_1) + Fr(r_0)\Delta Q'_{i+N/4}(R_1)\\ &+ Q'_{i+N/4}(R_1) + \gamma_1\Delta Q'_{i+N/4}(R_1)\end{aligned} \tag{13}$$

where

$$\gamma_1 = \begin{cases} 0 & \text{if } 0 \le Fr(r_0) + Fr(r_2) < 1 \\ -1 & \text{if } 1 \le Fr(r_0) + Fr(r_2) < 2. \end{cases}$$

Similarly, $\bar{q}_{i,3}(x,y)$ can be written as (compare to (13))

$$\begin{aligned}\bar{q}_{i,3}(x,y) =& Fr(r_2)\Delta Q'_{i+3N/4}(R_3) - Fr(r_0)\Delta Q'_{i+3N/4}(R_3)\\ &+ Q'_{i+3N/4}(R_3) + \gamma_3\Delta Q'_{i+3N/4}(R_3)\end{aligned} \tag{14}$$

where $R_3 = [-r_0 + r_2]$ and

$$\gamma_3 = \begin{cases} 1 & \text{if } 0 \le -Fr(r_0) + Fr(r_2) < 1 \\ 0 & \text{if } -1 < -Fr(r_0) + Fr(r_2) < 0. \end{cases}$$

Considering (7), (8), (13), and (14), we find that $\sum_{j=0}^{3}\bar{q}_{i,j}(x,y)$ can be computed using 2 multiplications, 2 if statements and an average of 16 additions, whereas the classical formula for linear interpolation-backprojection requires 4 multiplications and 12 additions (excluding integer additions for both cases).

The basic symmetry incorporated in the algorithm is shown in Fig. 1, where we consider the contribution of views numbered $1, N/4+1$ and $N/2+1$ to the reconstruction of a point P. Vertical, horizontal and diagonal lines show, respectively, the points where the backprojections of views $1, N/2+1$ and $N/4+1$ are known without need for interpolation. Notice that the spacing between vertical and horizontal lines is 1, whereas the spacing between diagonal lines is $\sqrt{2}/2$. If P is in the triangle ABC, then $Fr(r_0) + Fr(r_2) < 1$ and we see from the figure that $Fr(r_1) = Fr(r_0) + Fr(r_2)$. Hence, $\bar{q}_{i,1}(x,y)$ in (11) can be expressed using $Fr(r_0)$ and $Fr(r_2)$. A similar argument applies when P is in the triangle BCD and $1 < Fr(r_0) + Fr(r_2) < 2$.

We now compare our algorithm to some of the existing fast backprojection algorithms. In [3], an incremental algorithm is used to reduce the number of multiplications to $O(NM)$ and the computation run-time is improved by a factor of 1.86 (or 4.43 depending on the processor) for a 127×127 image. However, for efficient implementation, [3] requires a complex search flow algorithm whereas ours does not. In [2], linograms are used to reduce the number of operations to $O(NM\log M)$ and a speed-up factor of 2 to 3 is reported. However, note that [2] requires the projections to be on a nonuniform grid in the (r,θ) plane. Finally, [4] reports a speed-up factor of 1.2 for a 128×128 image; their speed-up factor varies proportionally to image size and is less than 1 for smaller images.

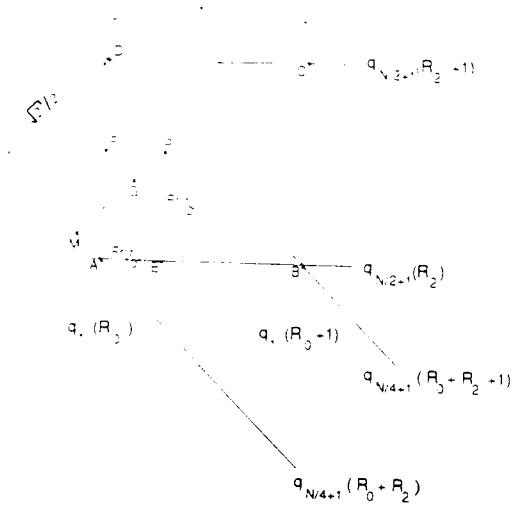


Fig. 1. Basic symmetry incorporated in the algorithm.

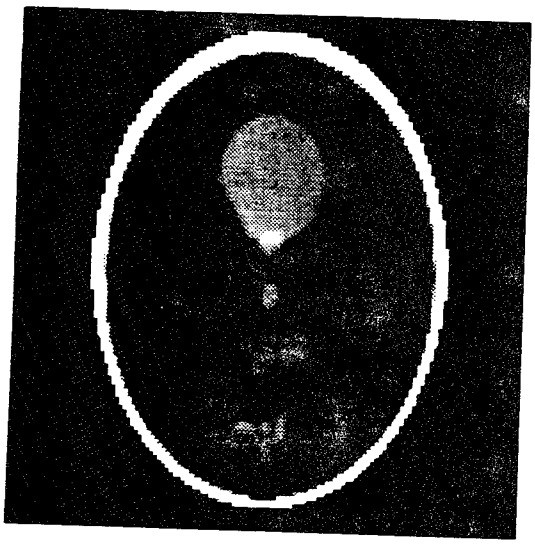


Fig. 2. Reconstructed image using classical FBP, windowed between 1.00 and 1.04.

III. Computation of $Q'(R)$ and Examples

In order to be able to implement (13) and (14), we need to resort to some kind of interpolation to compute $Q'(R)$ from $Q(R)$ (in the sequel, for simplicity, we drop the subscript i). Many interpolation methods for computing $Q'(R)$, such as bandlimited, spline, or Lagrange interpolation, will require far fewer operations than $O(NL^2)$; hence the computational overhead will be small. In this section, we discuss linear and bandlimited interpolation and give examples.

Linear Interpolation

Linear interpolation of $Q(R)$ to obtain $Q'(R)$ is very simple and requires only about $\frac{N}{2}M\sqrt{2}$ multiplications. The resulting algorithm will not be completely equivalent to conventional FBP, due to the distortions caused by linear interpolation. However, computed images are indistinguishable. Fig. 2 shows the result of applying the conventional FBP [6] to the Shepp-Logan phantom of [6], and Fig. 3 shows the result of applying our new algorithm using linear interpolation.

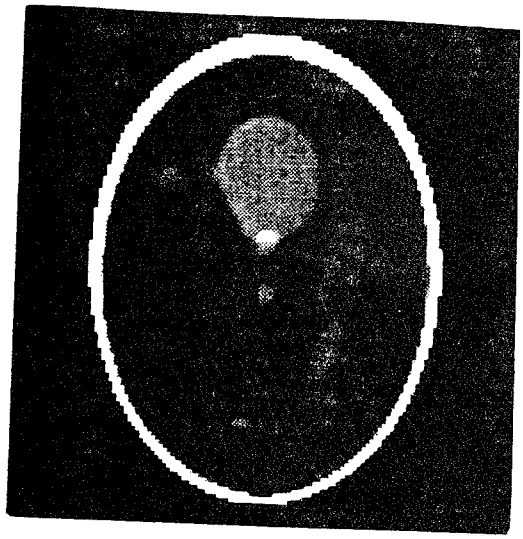


Fig. 3. Reconstructed image using the new algorithm with linear interpolation.

Bandlimited Interpolation

Bandlimited interpolation can be performed using the chirp z-transform (CZT) [5]. For an interpolation function with few ripples, we use a technique described in detail in [4]. Specifically, we let

$$Q'(k) = \sum_{R=0}^{M-1} Q(R)e^{-j\frac{2\pi}{M}k(R+M/2-1)}, k = -M, \cdots, M-1 \quad (15)$$

and compute

$$Q'(R) = \frac{F(k)}{M}\sum_{k=-M}^{M-1} Q'(k)e^{j\frac{2\pi}{M\sqrt{2}}Rk}, R = 0, \cdots, \lfloor (M-1)\sqrt{2} \rfloor \quad (16)$$

where $F(k)$ is a half-band filter, for example $F(k) = 0.5(1+\cos \pi k/M)$. (16) can be computed using the CZT. The additional computation required for this interpolation is one M-point FFT for (15) and two $(\sqrt{2}+2)M$ FFTs to implement (16). The overall additional cost is again much smaller than NL^2 for a typical image for which $L \approx M$. The result of our new algorithm using bandlimited interpolation is shown in Fig. 4.

IV. Generalization to More Than Four Views

Our algorithm uses the fact that views $i, i+N/4, i+N/2$ and $i+3N/4$ can be backprojected together. In this section, we generalize our idea to backproject more than four views together.

Consider the backprojection of a view at an angle θ_l, where $\theta_l = \theta_i + \Delta\theta$. The contribution of this view to the reconstructed image is

$$c_l(x, y) = \frac{\pi}{N}q_l(x\cos\theta_j + y\sin\theta_j) = \frac{\pi}{N}q_l(r_0\cos\Delta\theta + r_2\sin\Delta\theta). \quad (17)$$

Let $\tan\Delta\theta = \frac{A}{B}$ where A and B are small integers, (e.g.

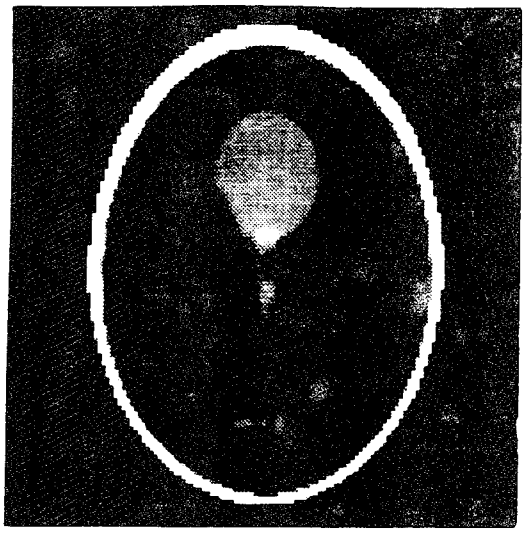


Fig. 4. Reconstructed image using the new algorithm with bandlimited interpolation.

$A, B \in \{-2, -1, 1, 2\}$), and let us write (17) as

$$c_l(x, y) = \frac{\pi}{N} q_l\left(\frac{\cos \Delta\theta}{B}(Br_0 + Ar_2)\right). \tag{18}$$

Repeating the same reasoning used in Section II, if we have samples of q_l with a sampling interval of $\left|\frac{\cos \Delta\theta}{B}\right|$, then we can interpolate

$$q_l\left(\frac{\cos \Delta\theta}{B}(Br_0 + Ar_2)\right)$$

using a multiplication by

$$\begin{aligned} Fr(Br_0 + Ar_2) &= Fr(BFr(r_0) + AFr(r_2)) \\ &= BFr(r_0) + AFr(r_2) - C \end{aligned} \tag{19}$$

where $C = [BFr(r_0) + AFr(r_2)]$.

Since they are small integers, multiplications by A, B, and C can be carried out using a few additions and if statements, so that the only multiplications in the computation of $c_l(x, y)$ will be by $Fr(r_0)$ and $Fr(r_2)$. Hence we can combine the backprojection of the angle θ_l with that of θ_i and $\theta_{i+N/2}$, factoring and saving the multiplications that would be required to backproject one view. Depending on the ratio of the addition/multiplication time or area of the processor, this may represent some savings in the computation time or chip area required.

Unfortunately, in a "natural" angular sampling scheme, angular increments are uniform, so that the only possible angles satisfying $\tan \Delta\theta = \frac{1}{B}$ are $\Delta\theta = \frac{\pi}{4}$ or $\Delta\theta = \frac{3\pi}{4}$. The idea of combining interpolation and backprojection of more than four views will be applicable only if angular sampling is done using a non-uniform $\Delta\theta$ or if angular interpolation is performed.

V. Conclusion

We have presented an algorithm which, by operating on 4 views together, saves 50% of multiplications in the linear interpolation and backprojection stage of the FBP method. Unlike the method suggested in [4] which uses an approximation to the trigonometric functions, this algorithm reduces the number of multiplications even for small M and L. The resulting image is virtually identical to that obtained by classical FBP.

We have also generalized the algorithm to backproject more than 4 views together. This generalized version, however, may increase the number of additions disproportionately, and requires nonuniformly spaced angular sampling.

References

[1] S. W. Rowland, "Computer implementation of image reconstruction formulas," in *Image Reconstruction from Projections: Implementation and Applications*, G. T. Herman, ed., vol. 32 of *Topics in Applied Physics*. New York: Springer-Verlag, 1979.

[2] P. Edholm, G. T. Herman, and D. A. Roberts, "Image reconstruction from linograms: Implementation and evaluation," *IEEE Trans. Medical Imaging*, vol. 7, pp. 239–246, 1988.

[3] Z. H. Cho, C. M. Chen, and S.-Y. Lee, "Incremental algorithm: A new fast backprojection scheme for parallel beam geometries," *IEEE Trans. Medical Imaging*, vol. 9, pp. 207–217, 1990.

[4] M. Tabei and M. Ueda, "Backprojection by upsampled Fourier series expansion and interpolated FFT," *IEEE Trans. Image Processing*, vol. 1, pp. 77–87, 1992.

[5] L. R. Rabiner, R. W. Schafer, and C. M. Rader, "The chirp z-transform algorithm and its application," *Bell Syst. Tech. J.*, vol. 48, pp. 1249–1272, 1969.

[6] L. A. Shepp and B. F. Logan, "The Fourier reconstruction of a head section," *IEEE Trans. Nucl. Sci.*, vol. NS-21, pp. 21–42, 1974.

Correction to "Adaptive Mathematical Morphology for Range Imagery"

Jacques G. Verly and Richard L. Delanoy

On page 274 of the above paper,[1] a portion of Figure 3 was mistakenly removed during the printing process. The complete figure is reproduced here.

Manuscript received July 8, 1993.

The authors are with the Machine Intelligence Technology Group, MIT Lincoln Laboratory, Lexington, MA 02173–9108.

IEEE Log Number 9213792.

[1] J. G. Verly and R. L. Delanoy, *IEEE Trans. Image Processing*, vol. 2, no. 2, Apr. 1993.

1057–7149/93$03.00 © 1993 IEEE

APPENDIX L2

B. Sahiner and A.E. Yagle, "A Fast Algorithm for Backprojection," IEEE 1992 Medical Imaging Conference, Orlando, FL, Oct. 25-31, 1992, pp. 1169-71.

This is the conference paper version of Appendix L1.

A FAST ALGORITHM FOR BACKPROJECTION

Berkman Sahiner and Andrew E. Yagle *
Department of Electrical Engineering and Computer Science
The University of Michigan, Ann Arbor, Michigan 48109-2122

Abstract

In the filtered backprojection procedure for image reconstruction from projections, backprojection dominates the computation time. We propose a simple algorithm which reduces the number of multiplications in linear interpolation and backprojection stage by 50%, with a small increase in the number of additions. The algorithm performs the interpolation and backprojection of four views together. Examples of implementation are given and extension to more than four views is discussed.

I. Introduction

The basic problem of x-ray tomography is to reconstruct an image $\mu(x,y)$ from its projections $p(r,\theta)$ where

$$p(r,\theta) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mu(x,y)\delta(r - x\cos\theta - y\sin\theta)dxdy \quad (1)$$

is the Radon transform of $\mu(x,y)$

The most common procedure for image reconstruction from projections is filtered backprojection (FBP), in which the reconstruction is carried out in two stages. The first stage is filtering, in which the projections are filtered by a modified ramp filter to yield the filtered projections $q(r,\theta)$ [1]. The second stage is backprojection, in which the desired image $\mu(x,y)$ is obtained from filtered projections by backprojection.

In practical problems, we have only samples of $p(r,\theta)$. Let us denote

$$P_i(R) = p(r,\theta_i)|_{r=Rd,\theta_i=\pi i/N}, \quad R = 0,\ldots,M-1, \quad i = 1,\ldots,N \quad (2)$$

where N is the total number of views, M is the number of samples in each view, and d is the radial sampling distance.

Similarly, let $Q_i(R)$ denote the discrete filtered projections, obtained by filtering $P_i(R)$ using a modified ramp filter. Then the backprojection operation is

$$\mu(x,y) = \frac{\pi}{N}\sum_{i=1}^{N} q_i(x\cos\theta_i + y\sin\theta_i) \quad (3)$$

where $q_i(r) = q(r,\theta_i)$ is obtained from $Q_i(R)$ by some kind of interpolation.

In the FBP method, backprojection dominates the computation time. The filtering stage can be done using the Fast Fourier Transform (FFT), which requires $O(NMlog_2M)$ operations. On the other hand, in the backprojection stage, interpolation needs to be done for every x, y, and i in (3); therefore if linear interpolation (requiring 1 multiplication per interpolated point) is used, then NL^2 multiplications and $3NL^2$ additions are needed to compute an $L \times L$ image.

In this paper, we propose a modification in the linear interpolation-backprojection stage which reduces the number of multiplications by about 50%, with a slight increase in the number of additions. This may result in a significant savings in computational time for processors which perform multiplications slower than additions. It will also result in reduced chip area in a VLSI implementation, since multipliers consume more chip area than adders.

In the next section, the basic idea is presented. In Section 3, we discuss some implementation issues and give examples. A generalization of the basic idea is presented in Section 4.

II. The Fast Algorithm

If linear interpolation is used in (3), then $q_i(r)$ is approximated as:

$$q_i(r) = Q_i(R) + Fr(r)\Delta Q_i(R), \quad (4)$$

where $R = [r]$ is the largest integer less than r, $Fr(r) = r - R$ is the fractional part of r, and the Δ operator acting on a discrete function is defined as $\Delta f(k) = f(k+1) - f(k)$.

We now show that if we backproject *four* views, namely the views numbered i, $i+N/4$, $i+N/2$, and $i+3N/4$, then multiplications by $Fr(r)$ above can be combined.

For notational convenience, let us denote

$$\bar{q}_{i,j}(x,y) = q_{i+\frac{N}{4}j}\left(r = x\cos(\theta_i + \tfrac{\pi}{4}j) + y\sin(\theta_i + \tfrac{\pi}{4}j)\right), \quad j = 0,1,2,3, \quad (5)$$

and rewrite (3) as

$$\mu(x,y) = \frac{\pi}{N}\sum_{i=1}^{N/4}\sum_{j=0}^{3} \bar{q}_{i,j}(x,y). \quad (6)$$

Let us explicitly write $\bar{q}_{i,0}(x,y)$ and $\bar{q}_{i,2}(x,y)$ in terms of the given filtered projections $Q_i(R)$ as

$$\bar{q}_{i,0}(x,y) = Fr(r_0)\Delta Q_i(R_0) + Q_i(R_0), \quad (7)$$

where $R_0 = [r_0]$ and $r_0 = x\cos\theta_i + y\sin\theta_i$. Similarly,

$$\bar{q}_{i,2}(x,y) = Fr(r_2)\Delta Q_{i+N/2}(R_2) + Q_{i+N/2}(R_2), \quad (8)$$

where $r_2 = -x\sin\theta_i + y\cos\theta_i$. We find that to compute $\bar{q}_{i,0}(x,y)$ and $\bar{q}_{i,2}(x,y)$, we need to perform two multiplications, one by $Fr(r_0)$ and another by $Fr(r_2)$.

*This work was supported by the Office of Naval Research under grant #N00014-90-J-1897

0-7803-0883-2/93/$3.00 ©1993 IEEE

Let us now express $\bar{q}_{i,1}(x,y)$ and $\bar{q}_{i,3}(x,y)$ using the same factors. Applying the sine and cosine addition formulas to (5) yields

$$\bar{q}_{i,1}(x,y) = q_{i+N/4}(r = \frac{\sqrt{2}}{2}(r_0 + r_2)). \quad (9)$$

If we have samples of $q_{i+N/4}$ with a sampling interval of $\sqrt{2}d/2$, i.e., if we have the discrete signal

$$Q'_i(R) = q_i(r)|_{r=Rd/\sqrt{2}}, \quad R = 0, 1, 2, \ldots, [M\sqrt{2}] \quad (10)$$

then $\bar{q}_{i,1}(x,y)$ can be expressed as

$$\bar{q}_{i,1}(x,y) = Fr(r_1)\Delta Q'_{i+N/4}(R_1) + Q'_{i+N/4}(R_1) \quad (11)$$

where $r_1 = r_0 + r_2$ and $R_1 = [r_1]$. However, using

$$Fr(r_1) = Fr(r_0 + r_2) = Fr(Fr(r_0) + Fr(r_2)), \quad (12)$$

equation (11) can be written as

$$\bar{q}_{i,1}(x,y) = Fr(r_2)\Delta Q'_{i+N/4}(R_1) + Fr(r_0)\Delta Q'_{i+N/4}(R_1) + Q'_{i+N/4}(R_1) + \gamma_1 \Delta Q'_{i+N/4}(R_1) \quad (13)$$

where

$$\gamma_1 = \begin{cases} 0 & \text{if } 0 \le Fr(r_0) + Fr(r_2) < 1 \\ -1 & \text{if } 1 \le Fr(r_0) + Fr(r_2) < 2. \end{cases}$$

Similarly, $\bar{q}_{i,3}(x,y)$ can be written as (compare to (13))

$$\bar{q}_{3}(x,y) = Fr(r_2)\Delta Q'_{i+3N/4}(R_3) - Fr(r_0)\Delta Q'_{i+3N/4}(R_3) + Q'_{i+3N/4}(R_3) + \gamma_3 \Delta Q'_{i+3N/4}(R_3) \quad (14)$$

where $R_3 = [-r_0 + r_2]$ and

$$\gamma_3 = \begin{cases} 1 & \text{if } 0 \le -Fr(r_0) + Fr(r_2) < 1 \\ 0 & \text{if } -1 < -Fr(r_0) + Fr(r_2) < 0. \end{cases}$$

Considering (7), (8) (13), and (14), $\sum_{j=0}^{3} \bar{q}_{i,j}(x,y)$ can be computed using 2 multiplications, 2 if statements and an average of 16 additions; the classical formula for linear interpolation-backprojection requires 4 multiplications and 3 additions (excluding integer additions for both cases).

III. Computation of $Q'(R)$ and Examples

To be able to implement (13) and (14), we need to resort to some kind of interpolation to compute $Q'(R)$ from $Q(R)$. (In the sequel, for simplicity, we drop the subscript $'$.) Many interpolation methods for computing $Q'(R)$, such as bandlimited, spline, or Lagrange interpolation, will require far fewer operations than $O(NL^2)$; hence the computational overhead will be small. In this section, we discuss linear and bandlimited interpolation; and give examples.

A. Linear Interpolation

Linear interpolation of $Q(R)$ to obtain $Q'(R)$ is very simple and requires only about $\frac{N}{2}M\sqrt{2}$ multiplications. Figure 1 shows the result of applying the conventional FBP [3] to the Shepp-Logan phantom of [3], and Figure 2 shows the result of applying our new algorithm using linear interpolation. The results are indistinguishable.

B. Bandlimited Interpolation

Bandlimited interpolation can be performed using the Chirp z-transform (CZT) [2]. Specifically, if $\hat{Q}(k), k = 0, \ldots, M-1$ is the discrete Fourier transform of $Q(R)$, then we let

$$\hat{Q}'(k) = \begin{cases} \frac{1}{2}\hat{Q}(k) & k = M/2 \\ \hat{Q}(k) & k = 0, \ldots, M/2 - 1 \\ \hat{Q}(M-k) & k = -1, \ldots, -M/2 + 1 \\ \frac{1}{2}\hat{Q}(M-k) & k = -M/2 \end{cases} \quad (15)$$

and compute

$$Q'(R) = \frac{1}{M} \sum_{k=-M/2}^{M/2} e^{j\frac{2\pi}{N\sqrt{2}}Rk} \hat{Q}'(k) \quad (16)$$

for $R = 0, \ldots, [(M-1)\sqrt{2}]$, which can be done using the CZT [2]. This corresponds to the interpolation

$$Q'(R) = \frac{1}{M} \sum_{m=0}^{M-1} Q(m) h(\frac{R}{\sqrt{2}} - m), \; h(x) = \frac{\sin \pi x}{\tan \frac{\pi x}{M}}. \quad (17)$$

Using the CZT, this procedure requires the computation of a single $M' = [M(\sqrt{2}+1)+1]$-point inverse FFT for each view for the convolution. ($\hat{Q}(k)$ is already available from the filtering stage.) This requires roughly $\frac{N}{2}M' \log_2 M'$ multiplications, which is usually much smaller than NL^2 for a typical image for which $L \approx M$.

Although bandlimited interpolation is attractive and simple to implement, it has a major drawback: The interpolation function $h(x)$ has too many ripples. These ripples can be reduced using a smoothing function in the frequency domain, but at the expense of spatial resolution. Figure 3 shows the result of applying our new algorithm with bandlimited interpolation. Notice that the ripples caused by the high-density skull area go deep inside the picture.

From this, we conclude that linear interpolation is faster, and seems to give better results.

IV. A Generalization

Our algorithm uses the fact that views $i, i+N/4, i+N/2$ and $i+3N/4$ can be backprojected together. In this section, we generalize our idea to backproject more than four views together.

Consider the backprojection of a view at an angle θ_l, where $\theta_l = \theta_i + \Delta\theta$. The contribution of this view to the reconstructed image is

$$c_l(x,y) = \frac{\pi}{N} q_l(x\cos\theta_j + y\sin\theta_j) = \frac{\pi}{N} q_l(r_0 \cos\Delta\theta + r_2 \sin\Delta\theta). \quad (18)$$

Let $\tan\Delta\theta = \frac{A}{B}$ where A and B are small integers, (e.g. $A, B \in \{-2, -1, 1, 2\}$), and let us write (18) as

$$c_l(x,y) = \frac{\pi}{N} q_l\left(\frac{\cos\Delta\theta}{B}(Br_0 + Ar_2)\right). \quad (19)$$

Repeating the reasoning used in Section 2, if we have samples of q_l with a sampling interval of $\left|\frac{\cos\Delta\theta}{B}\right|$, then we can interpolate $q_l(\frac{\cos\Delta\theta}{B}(Br_0+Ar_2))$ using a multiplication by

$$\begin{aligned} Fr(Br_0+Ar_2) &= Fr(BFr(r_0)+AFr(r_2)) \\ &= BFr(r_0)+AFr(r_2)-C, \end{aligned} \quad (20)$$

where $C=[BFr(r_0)+AFr(r_2)]$.

Since A, B, and C are small integers, multiplications by these integers can be carried out using a few additions and if statements, so that the only multiplications in the computation of $c_l(x,y)$ will be by $Fr(r_0)$ and $Fr(r_2)$. Hence we can combine the backprojection of the angle θ_l with that of θ_i and $\theta_{i+N/2}$, factoring and saving the multiplications that would be required to backproject one view. Depending on the ratio of the addition/multiplication time or area of the processor, this may represent some savings in the computation time or chip area required.

Unfortunately, in a "natural" angular sampling scheme, angular increments are uniform, so that the only possible angles satisfying $\tan\Delta\theta=\frac{A}{B}$ are $\Delta\theta=\frac{\pi}{4}$ or $\Delta\theta=\frac{3\pi}{4}$. The idea of combining interpolation and backprojection of more than four views will be applicable only if angular sampling is done using a non-uniform $\Delta\theta$ or if angular interpolation is performed.

V. Conclusion

We have presented an algorithm which, by operating on 4 views together, saves 50% of multiplications in the linear interpolation and backprojection stage of the FBP method. Unlike the method suggested in [4] which uses an approximation to the trigonometric functions, this algorithm reduces the number of multiplications even for small M and L. The resulting image is virtually identical to that obtained by classical FBP.

We have also generalized the algorithm to backproject more than 4 views together. This generalized version, however, may increase the number of additions disproportionately, and requires non-uniformly spaced angular sampling.

References

[1] S. W. Rowland, "Computer Implementation of Image Reconstruction Formulas," in *Image Reconstruction from Projections*, G. T. Herman, Ed., vol. 32 of *Topics in Applied Physics*, New York: Springer-Verlag, 1979.

[2] L. R. Rabiner, R. W. Schafer, and C. M. Rader, "The Chirp z-Transform Algorithm and Its Application," Bell Syst. Tech. J., vol. 48, pp. 1249-1272, 1969.

[3] L. A. Shepp and B. F. Logan, "The Fourier reconstruction of a head section," IEEE Trans. Nucl. Sci., vol. NS-21, pp. 21-42, 1974.

[4] M. Tabei and M. Ueda, "Backprojection by Upsampled Fourier Series Expansion and Interpolated FFT," IEEE Trans. Image Proc., vol. 1, pp. 77-87, 1992.

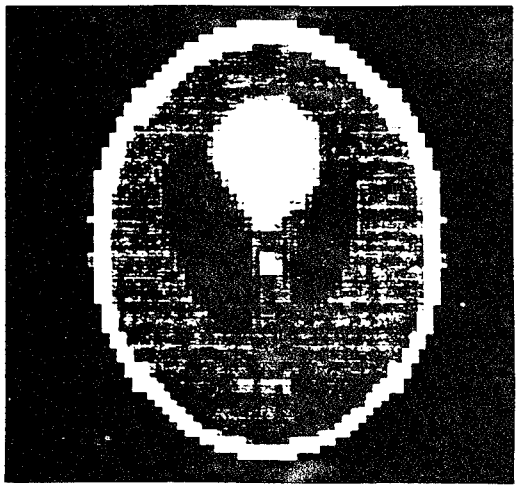


Figure 1: Reconstructed image using classical FBP, windowed between 1.00 and 1.04.

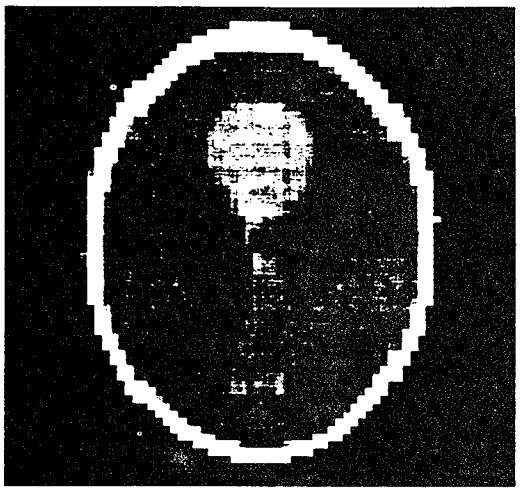


Figure 2: Reconstructed image using the new algorithm with linear interpolation, windowed between 1.00 and 1.04.

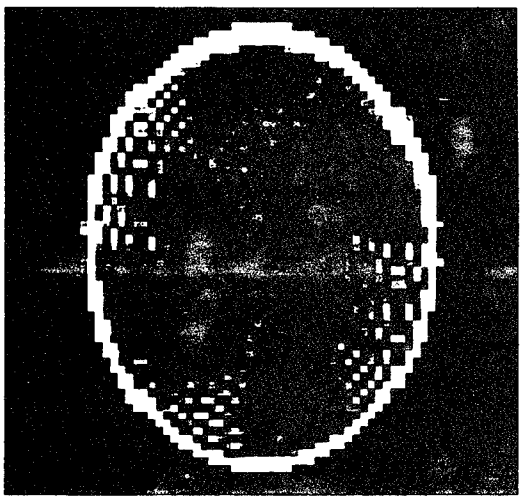


Figure 3: Reconstructed image using the new algorithm with bandlimited interpolation, windowed between 1.00 and 1.06

APPENDIX M

B. Sahiner and A.E. Yagle, "Limited Angle Tomography Using the Wavelet Transform," revision submitted to IEEE Trans. Image Proc., October 1993.

This paper shows that when the extent of missing angles is small in limited-angle tomography, two of the three sets of detail images in the wavelet transform are unaffected, and low-resolution images can be obtained by interpolation. Using some a priori partial information on edges parallel to the missing angles, we have developed a wavelet-domain algorithm for restoring the image.

LIMITED ANGLE TOMOGRAPHY USING THE WAVELET TRANSFORM

Berkman Sahiner and Andrew E. Yagle

Dept. of Electrical Engineering and Computer Science

The University of Michigan, Ann Arbor, Michigan 48109

Abstract

We investigate the problem of limited angle tomography when there exists a priori knowledge about edges that lie parallel to missing view angles. We fill the missing high-frequency regions in the Fourier plane using the edge knowledge, and the low-frequency regions using a simple interpolation procedure. We characterize the edges, and bring together the lowpass and highpass images by using the wavelet transform.

I. INTRODUCTION

The limited angle problem in tomography has received a lot of attention in recent years due to its applications in medical imaging, astronomy, electron microscopy, industry, etc. [1, 2]. One of the most important tools in limited angle reconstruction is a priori information about the image, which may be in the form of limited spatial extent of the image, upper and lower bounds on reconstructed pixel values, energy constraints, or closeness to a known image [3, 4]. In this correspondence, we investigate the problem of image restoration when the a priori knowledge is the existence of edges that lie parallel to the missing view angles.

To define the limited angle problem, we first review the use of complete data in tomography. The data consist of line integrals $p_f(r_m, \theta_n)$ of an image $f(x, y)$, which are defined as

$$p_f(r_m, \theta_n) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y)\delta(r_m - x\cos\theta_n - y\sin\theta_n)dxdy \tag{1}$$

where $\theta_n = (n + 0.5)\pi/N$, $n = 0, 1, ..., N-1$ and $r_m = \Delta r(m - M/2), m = 0, 1, \ldots, M-1$. It is assumed that Δr, M and N have been chosen appropriately to obtain an accurate reconstruction $f_r(i_1, i_2)$ using filtered backprojection. The filtered backprojection equation is

$$f_r(i_1, i_2) = \sum_{n=0}^{N-1} q_n(i_1\cos\theta_n + i_2\sin\theta_n) \tag{2}$$

EDICS Number: 6.3.1. Phone:313-763-9810

where $q_n(r)$ is obtained by convolving $p_f(r_m, \theta_n)$ with a ramp or modified ramp function [5].

We now define the limited angle problem. We assume, without loss of generality, that $2L$ views centered around $\pi/2$ are missing in the projection data, corresponding to a missing angle extent of $\theta_{miss} = \frac{2\pi L}{N}$. The available data are thus $p_f(r_m, \theta_n), n = 0, \ldots, \frac{N}{2} - L - 1, \frac{N}{2} + L, \ldots, N - 1$. The projection-slice theorem states that the Fourier transform $P_f(w, \theta_n)$ of $p_f(r_m, \theta_n)$ is equal to the Fourier transform $F(w_1, w_2)$ of the image $f(x, y)$ along a slice in the Fourier plane that passes through the origin and makes an angle θ_n with the w_1 axis. Thus, the Fourier transform of the image is known on the concentric circles grid shown in Figure 1, except for the bowtie-like region R. The quality of reconstruction from missing views depends on how well we can fill in the missing Fourier transform samples in R. It has recently been shown [6] that the squashing algorithm [7] is equivalent to setting frequency samples in R to zero. Oskoui and Stark have used an interpolation technique to fill R [8], however, results indicate that their interpolation does not improve over squashing. Still another method is projections onto convex sets (POCS), where the convex sets are defined by the given projections and the a priori knowledge about the image [9, 3]. These methods have been compared in [8].

In this correspondence, we propose to fill the high-frequency regions of R using knowledge of edges which lie parallel to the x axis, and to fill the low-frequency regions of R using a simple interpolation technique. We characterize the edges, and bring together the lowpass and highpass regions of R, by using the wavelet transform.

This correspondence makes four contributions: (1) We give a new interpretation of why edges in certain directions cause artifacts in their vicinity and are blurred, while other edges are not. (2) We give a new interpolation procedure to obtain a low-resolution image; this isolates directional high-resolution images as missing information. (3) With the use of a priori edge information to complete the reconstruction, we not only sharpen the edges but also eliminate the artifacts caused by these edges. (4) We present a number of numerical examples with two different phantoms, different wavelet bases, and different number of scales in the wavelet representation.

In Section II, we review the wavelet transform and show that if θ_{miss} is small enough, then two of the three sets of detail images in the wavelet transform will be largely unaffected. This provides a new explanation of why edges orthogonal to the missing view angles are reconstructed well, while edges parallel to the missing angles are blurred. We use the edge information in the remaining direction to partly fill in the affected detail image. A recent publication [10] has shown that it is possible to reconstruct an image from the values and location of the maxima of the wavelet transform, which roughly correspond

to the location of the edges. In our case, if the edges parallel to the x-axis are known, and the detail images in diagonal and y-axis directions are unaffected, it is not surprising that the original image can be recovered. In Section III, we discuss the interpolation technique that we use to fill in the lowpass regions of R. In Section IV, we present our algorithm and give numerical examples.

II. THE WAVELET TRANSFORM

In this correspondence, we restrict attention to discrete dyadic wavelet transforms. For a 1-D signal $f(i)$, we use the following definition for the wavelet transform, which is described in more detail in [10]:

$$\begin{aligned} S_1 f(i) &= f(k), \\ W_{2^{j+1}} f(i) &= S_{2^j} f(i) * G_j(i), \quad 0 \le j \le J, \\ S_{2^{j+1}} f(i) &= S_{2^j} f(i) * H_j(i), \quad 0 \le j \le J. \end{aligned} \tag{3}$$

At each scale j, the algorithm decomposes $S_{2^j} f$ into a detail signal $W_{2^{j+1}}$ and an average signal $S_{2^{j+1}}$. The filters $H_j(i)$ and $G_j(i)$ are obtained by inserting $2^j - 1$ zeros between the coefficients of $H_0(i)$ and $G_0(i)$, which means that their Fourier transforms satisfy $H_j(w) = H_0(2^j w)$ and $G_j(w) = G_0(2^j w)$. $H_0(w)$ is a lowpass filter with $H_0(0) = 1$ and $G_0(w)$ is a highpass filter with $G_0(0) = 0$. Note that unlike the definitions given in [11] or [12], this definition of the wavelet transform is redundant. This redundancy is useful when the purpose of the wavelet transform is characterization or representation of sharp transitions in the signal.

If $K(w)$ is a filter that satisfies $K(w)G(w) + |H(w)|^2 = 1$, then a perfect reconstruction from the wavelet transform is given by

$$S_{2^{j-1}} f(i) = W_{2^j} f(i) * K_{j-1}(i) + S_{2^j} f(i) * \tilde{H}_{j-1}(i) \tag{4}$$

where $H_{j-1}(i) = H(-2^{j-1} i)$ and $K_{j-1}(i) = K(2^{j-1} i)$.

In this paper, we choose to use $G(i) = (-1)^i H(1 - i)$ and $K(i) = G(-i)$, which implies that

$$|H(w)|^2 + |H(w + \pi)|^2 = 1. \tag{5}$$

Filters that satisfy (5) are called quadrature mirror filters; orthogonal wavelets are a special class of quadrature mirror filters.

For a 2-D signal $f(i_1, i_2)$, we define the wavelet transform as

$$\begin{aligned}
W_{2^{j+1}}^{(1)} f(i_1, i_2) &= S_{2^j} f(i_1, i_2) * G_j(i_1) * H_j(i_2) \quad 0 \le j \le J, \\
W_{2^{j+1}}^{(2)} f(i_1, i_2) &= S_{2^j} f(i_1, i_2) * H_j(i_1) * G_j(i_2) \quad 0 \le j \le J, \\
W_{2^{j+1}}^{(3)} f(i_1, i_2) &= S_{2^j} f(i_1, i_2) * G_j(i_1) * G_j(i_2) \quad 0 \le j \le J, \\
S_{2^{j+1}} f(i_1, i_2) &= S_{2^j} f(i_1, i_2) * H_j(i_1) * H_j(i_2) \quad 0 \le j \le J.
\end{aligned} \tag{6}$$

If H satisfies (5), and with the choice of $G(i) = (-1)^i H(1-i)$, the reconstruction algorithm is

$$\begin{aligned}
S_{2^j} f(i_1, i_2) &= W_{2^j}^{(1)} f(i_1, i_2) * \tilde{G}_j(i_1) * \tilde{H}_j(i_2) + W_{2^j}^{(2)} f(i_1, i_2) * \tilde{H}_j(i_1) * \tilde{G}_j(i_2) \\
&+ W_{2^j}^{(3)} f(i_1, i_2) * \tilde{G}_j(i_1) * \tilde{G}_j(i_2) + S_{2^j} f(i_1, i_2) * \tilde{H}_j(i_1) * \tilde{H}_j(i_2).
\end{aligned} \tag{7}$$

The ideal partitioning of the Fourier plane with the filters H and G is shown in Figure 2. If H and G are ideal lowpass and highpass filters, and $\theta_{miss} \le 2\tan^{-1}\frac{1}{3} = 36.8^o$, then we observe from Figure 2 that $W_{2^j}^{(1)} f(i_1, i_2)$ and $W_{2^j}^{(3)} f(i_1, i_2)$ will not be affected. This is a new explanation of why edges not parallel to the missing views are relatively unaffected by them.

As long as θ_{miss} is not larger than 36.8^o, $W_{2^j}^{(2)} f(i_1, i_2)$ will be affected, but will still carry information about the image. In fact, the missing information lost along with the missing angles shows up in the effect on $W_{2^j}^{(2)} f(i_1, i_2)$. Specification of the missing data in Fourier space, which is global, can thus be replaced with specification of the missing data in the local, directional wavelet coefficients $W_{2^j}^{(2)} f(i_1, i_2)$.

Edges in the image $f(i_1, i_2)$ correspond to high frequency components in the Fourier domain. However, they can also be viewed as localized high-resolution components in the wavelet domain. In a limited angle tomography problem, if the locations and magnitudes of some of the edges that lie parallel to the x-axis in the image are known, then $W_{2^j}^{(2)} f(i_1, i_2)$ can be approximated up to some scale J for values of i_1 and i_2 near the known edges. However, since the edges give us little information about the low-resolution behavior of the image, a different method must be used to fill this low-resolution information, which corresponds to lowpass regions in the Fourier plane. A simple algorithm for this is given is Section III.

If $\theta_{miss} < 36.8^o$ or if the filter H is not ideal, all three detail images in the wavelet transform of the image obtained from the missing data will contain useful information; therefore we need not throw away one detail image completely and replace it with the approximation obtained by the edge information. This problem is solved by the use of forward projection in Section IV.

III. INTERPOLATION OF LOWPASS MISSING DATA

In the Fourier plane, the projections provide us with Fourier values on a concentric circles grid, which does not have uniform sampling density. The sampling density is highest around the origin, and decreases linearly as the frequency magnitude increases. This is one reason why a ramp filter is needed in the convolution-backprojection method— high-frequency data are sampled more sparsely, so each datum must be weighted more heavily. On the other hand, if the image $f(x, y)$ is spatially limited, as is the case for many tomography problems, its Fourier transform $F(w_1, w_2)$ can be interpolated using its samples on a rectangular grid which has uniform density everywhere. Although the ideal interpolation function here is a sinc, which has infinite support, in most cases a few neighboring samples are sufficient for interpolation. This difference in sampling densities suggests that there may be a high correlation among the low frequency samples on a concentric circles grid.

As an illustrative example, assume that $f(x, y)$ is spatially limited to a circle of radius T, and assume that the radial sampling interval for the concentric circles grid in the Fourier plane is the Nyquist interval π/T. Since $f(x, y)$ is spatially limited, it is not frequency-limited; therefore, strictly speaking, we need an infinite number of samples per radial line for exact representation. However, usually a finite number of samples M is sufficient. Also, assume that the number of angular samples N is equal to M, giving us the grid shown in Figure 3.

According to the sampling theorem, $F(w_1, w_2)$ can be interpolated in terms of its samples $F(\frac{k_1\pi}{T}, \frac{k_2\pi}{T})$. Assume that we use the nearest neighbors in a $\frac{\Delta\pi}{T} \times \frac{\Delta\pi}{T}$ square centered around (w_1, w_2) to interpolate $F(w_1, w_2)$, where Δ is an integer. Then, to interpolate the Fourier transform in a square $\frac{K\pi}{T} \times \frac{K\pi}{T}$, we will need $(K + \Delta)^2$ samples. However, in the circle of radius $\frac{\pi K}{2T}$ inscribed in the $\frac{K\pi}{T} \times \frac{K\pi}{T}$ square, note that the concentric circles grid has NK samples. This means that for every sample point necessary for interpolation in the rectangular grid, there are $\frac{NK}{(K+\Delta)^2}$ samples in the concentric circles grid. If $N = M = K \gg \Delta$ then this ratio is 1, however, if $K = M/16 \gg \Delta$, then this ratio is 16. This means that the sampling is redundant; some of these samples can be interpolated from others.

Although a number of interpolation methods could be used, we chose the following simple linear interpolation procedure:

$$P_f(w_m, \theta_n) = \frac{1}{2L+1}\left[\left(\frac{N}{2} + L - n\right) P_f(w_m, \theta_{N/2-L-1}) + \left(n - (\frac{N}{2} - L - 1)\right) P_f(w_m, \theta_{N/2+L})\right] \quad (8)$$

where the Fourier transform of any missing view in the range of missing views is replaced by a linear combination Fourier transforms of the two views which border to the missing view range.Note that since

the interpolation is linear, missing projection data can be interpolated simply by taking the inverse Fourier transform of (8), i.e., by replacing w_m in (8) by r_m.

To test the validity of this procedure, we tried it on two test images. The first image consists of three geometric shapes. The second image is the Shepp-Logan phantom, in which the gray levels are chosen as in [13]; this is a frequently-used phantom in limited angle studies [3, 8]. The reconstructed images $f_r(i_1, i_2)$ from full views with $N = M = 128$ are shown in Figures 4 and 5. Table 1 compares the images obtained using the interpolation formula (8) to the images obtained by setting the unknown Fourier coefficients to zero, as in the squashing algorithm [6]. The number of missing views $2L$ is 16 for the geometric phantom and 32 for the Shepp-Logan phantom. The basis for comparison is the percent root mean square error.

$$E_{rms} = (100\%)\frac{\|\tilde{f} - f_r\|_2}{\|f_r\|_2}. \tag{9}$$

The first row in Table 1 shows the error for f_{lim}, obtained by setting the unknown Fourier coefficients to zero, and the second row shows the error for f_{pl}, obtained by using the interpolation formula (8). The third and fourth rows show the error for $S_{2^4} f_{lim}$ and $S_{2^4} f_{pl}$ computed using the Haar basis. It is observed that although the interpolation does not result in a dramatic improvement over the image itself, the improvement in the lowpass signal $S_{2^4} f$ is dramatic.

IV. THE ALGORITHM AND NUMERICAL EXAMPLES

We summarize below our algorithm for limited angle tomographic reconstruction, given the locations and magnitudes of edges that lie parallel to missing view angles:

1. Interpolate unknown projections from the known ones using (8). As discussed in Section III, this interpolation will work well at low frequencies but not so well at high frequencies.

2. Reconstruct an image $f_{pl}(i_1, i_2)$ from the known and interpolated views using filtered backprojection.

3. Find the wavelet transform of $f_{pl}(i_1, i_2)$ up to some scale J using (6).

4. From the knowledge of edges that lie parallel to i_1 axis, construct an image $E(i_1, i_2)$ that has edges of known magnitude at known locations, and no edges elsewhere. Compute $W_{2^j}^{(2)} E(i_1, i_2), j = 1, \ldots, J$.

5. Reconstruct an image $f_{epl}(i_1, i_2)$ using the inverse wavelet transform from the images $\{W_{2^j}^{(1)} f_{pl}(i_1, i_2), W_{2^j}^{(2)} E(i_1, i_2), W_{2^j}^{(3)} f_{pl}(i_1, i_2), j = 1, \ldots, J\}$ and $S_{2^J} f_{pl}(i_1, i_2)$.

6. Compute projections $p_{f_{epl}}(r_m, \theta_n)$ of $f_{epl}(i_1, i_2)$ along the missing views $\theta_n = \theta_{N/2-L}, \ldots, \theta_{N/2+L-1}$ to complete the missing projection data. The completed set of views $\hat{p}_f(r_m, \theta_n)$ is

$$\hat{p}_f(r_m, \theta_n) = \begin{cases} p_{f_{epl}}(r_m, \theta_n) & \text{if} \quad \theta_{N/2-L} \leq \theta_n \leq \theta_{N/2+L-1} \\ p_f(r_m, \theta_n) & \text{otherwise.} \end{cases} \tag{10}$$

Finally, use the filtered backprojection algorithm on $\hat{P}_f(r_m, \theta_n)$ to find the restored image $\hat{f}(i_1, i_2)$.

We now present numerical examples involving the geometric phantom of Figure 4 and the Shepp-Logan phantom of Figure 5. In all examples, complete data consist of 128×128 projections and the missing data are centered around $\pi/2$.

Figure 6 shows $f_{lim}(i_1, i_2)$, obtained using filtered backprojection when 16 views ($L = 8$, $\theta_{miss} = 22.5^\circ$) are missing from the projections of the geometric phantom. The missing views are replaced by zero in filtered backprojection, as in the squashing algorithm [6]. We observe that there is blurring and artifacts around the edges which lie parallel to the i_1 axis. Figure 7 shows the locations where we assume a priori edge knowledge. Figure 8 shows the edge image, $E(i_1, i_2)$ obtained from these locations and the edge magnitudes. Figure 9 shows $f_{epl}(i_1, i_2)$, reconstructed from $\{W_{2^j}^{(1)} f_{pl}(i_1, i_2), W_{2^j}^{(2)} E(i_1, i_2), W_{2^j}^{(3)} f_{pl}(i_1, i_2), j = 1, \ldots, 4\}$ and $S_{2^4} f_{pl}(i_1, i_2)$ where the wavelet basis for the decomposition and reconstruction was chosen as the Haar basis. Figure 10 shows the final restored image, $\hat{f}(i_1, i_2)$. We note that not only the edges parallel to the i_1 axis are now unblurred (this is natural because we *assumed* that we knew the existence of these edges), but also the artifacts around the edges have been significantly reduced.

Figure 11 shows $f_{lim}(i_1, i_2)$, for the Shepp-Logan phantom, with $L = 16$. Figure 12 shows the locations where we assume a priori edge knowledge, and Figure 13 shows the final restored image, $\hat{f}(i_1, i_2)$. Again, artifacts have almost completely been eliminated.

To quantify our results, we again use the percent root mean square error, defined in Section III. Table 2 summarizes the error for $\hat{f}$ with three wavelet bases $D2$ (Haar basis), $D4$ and $D6$ defined in [12] and for J=3, 4, 5, and 6. We observe that a short wavelet basis such as $D2$ or $D4$, and a moderate number of scales such as $J = 3$ or $J = 4$, are sufficient to obtain good restoration.

REFERENCES

[1] J. S. Jaffe, "Limited angle reconstruction using stabilized algorithms," IEEE Trans. Med. Imaging, vol. MI-9, pp. 338-344, 1990.

[2] K. C. Tam, J. W. Eberhard and K. W. Mitchell, "Incomplete-data image reconstructions in industrial applications," IEEE Trans. Nucl. Sci., vol. NS-37, pp. 1490-1499, 1990.

[3] P. Oskoui-Fard and H. Stark, "Tomographic image reconstruction using the theory of convex projections," IEEE Trans. Med. Imaging, vol. MI-7, pp. 45-59, 1988.

[4] M. I. Sezan and H. Stark, "Tomographic image reconstruction from incomplete data by convex projections and direct Fourier inversion," IEEE Trans. Med. Imaging, vol. MI-3, pp.91-98, 1984.

[5] L. A. Shepp and B. F. Logan, "The Fourier reconstruction of a head section," IEEE Trans. Nucl. Sci., vol. NS-21, pp. 21-42, 1974.

[6] T. Olson and J. S. Jaffe, "An explanation of the effects of squashing in limited angle tomography," IEEE Trans. Med. Imaging, vol. MI-9, pp. 242-246, 1990.

[7] J. A. Reeds and L. A. Shepp, "Limited angle reconstruction in tomography via squashing," IEEE Trans. Med. Imaging, vol. MI-6, pp. 89-97, 1987.

[8] P. Oskoui and H. Stark, "A comparative study of three reconstruction methods for a limited-view computer tomography problem," IEEE Trans. Med. Imaging, vol. MI-8, pp. 43-49, 1989.

[9] M. I. Sezan and H. Stark, "Image restoration by the method of convex projections: Part II," IEEE Trans. Med. Imaging, vol. MI-1 pp.95-101, 1982.

[10] S. Mallat and S. Zhong, "Characterization of signals from multiscale edges," IEEE Trans. Patt. Anal. Machine Intell., vol. PAMI-14, pp. 710-732, 1992.

[11] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet transform," IEEE Trans. Patt. Anal. Machine Intell., vol. PAMI-11, pp. 674-693, 1989.

[12] I. Daubechies, "Orthonormal bases of compactly supported wavelets," Commun. Pure Appl.Math., vol. 41, pp. 909-996, 1988.

[13] S. X. Pan and A. C. Kak, "A computational study of reconstruction algorithms for diffraction tomography: Interpolation versus filtered backpropagation," IEEE Trans. Acoust., Speech, Sig. Proc., vol. ASSP-31, pp. 1262-1275, 1983.

	Geometric phantom (Fig. 4) with 16 views missing	Shepp-Logan phantom (Fig. 5) with 32 views missing
E_{rms} for f_{lim}	38.96	28.91
E_{rms} for f_{pl}	22.76	16.82
E_{rms} for $S_{2^4} f_{lim}$	29.13	24.68
E_{rms} for $S_{2^4} f_{pl}$	6.03	3.87

Table 1. Comparison of E_{rms} between images obtained using interpolation and images obtained by setting unknown Fourier coefficients to zero.

	E_{rms} for $\hat{f}$ using											
	Haar basis				D2 basis				D4 basis			
	$J=3$	$J=4$	$J=5$	$J=6$	$J=3$	$J=4$	$J=5$	$J=6$	$J=3$	$J=4$	$J=5$	$J=6$
Geom. phantom	11.98	10.07	11.73	12.88	12.73	9.89	11.86	13.24	13.09	9.86	11.93	13.35
S-L phantom	7.08	6.30	7.18	8.98	7.30	6.35	7.06	9.03	7.40	6.42	7.08	9.01

Table 2. E_{rms} for different wavelet bases and different J.

Figure Headings

Fig. 1. The concentric circles grid and the bowtie-like region R over which Fourier transform samples are missing.

Fig. 2. The ideal partitioning of the Fourier plane with the filters H and G.

Fig. 3. The concentric circles grid and the rectangular grid for $M = N = 15$. Note that inside the small square in the center, the concentric grid has 30 samples and the rectangular grid has 9 samples.

Fig. 4. The geometric phantom, reconstructed from 128×128 projections.

Fig. 5. The Shepp-Logan phantom, reconstructed from 128×128 projections.

Fig. 6. $f_{lim}(i_1, i_2)$ obtained from 16 missing view angles ($\theta_{miss} = 22.5^o$).

Fig. 7. Locations of edges that are known to be parallel to the i_1 axis

Fig. 8. The edge image $E(i_1, i_2)$.

Fig. 9. $f_{epl}(i_1, i_2)$

Fig. 10. The final reconstructed image $\hat{f}(i_1, i_2)$.

Fig. 11. $f_{lim}(i_1, i_2)$ obtained from 32 missing view angles ($\theta_{miss} = 45^o$).

Fig. 12 Locations of edges that are known to be parallel to the i_1 axis

Fig. 13. The final reconstructed image $\hat{f}(i_1, i_2)$.

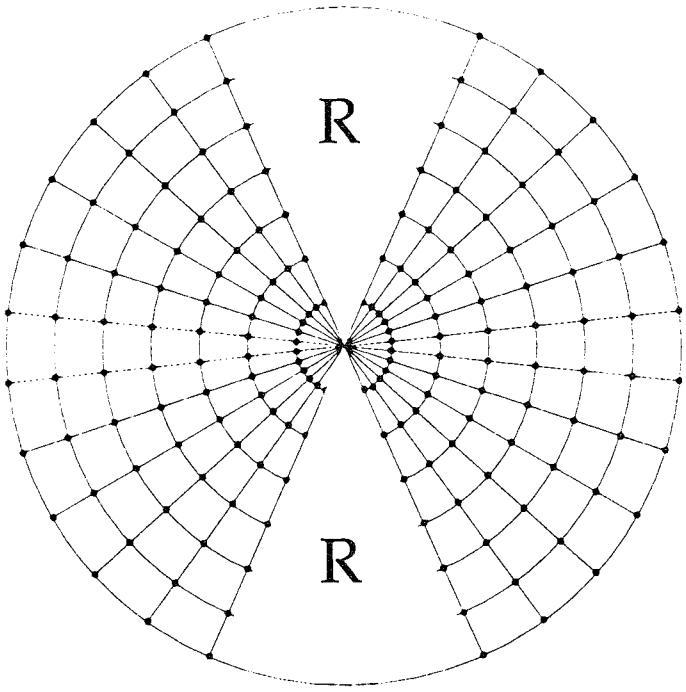


Figure 1

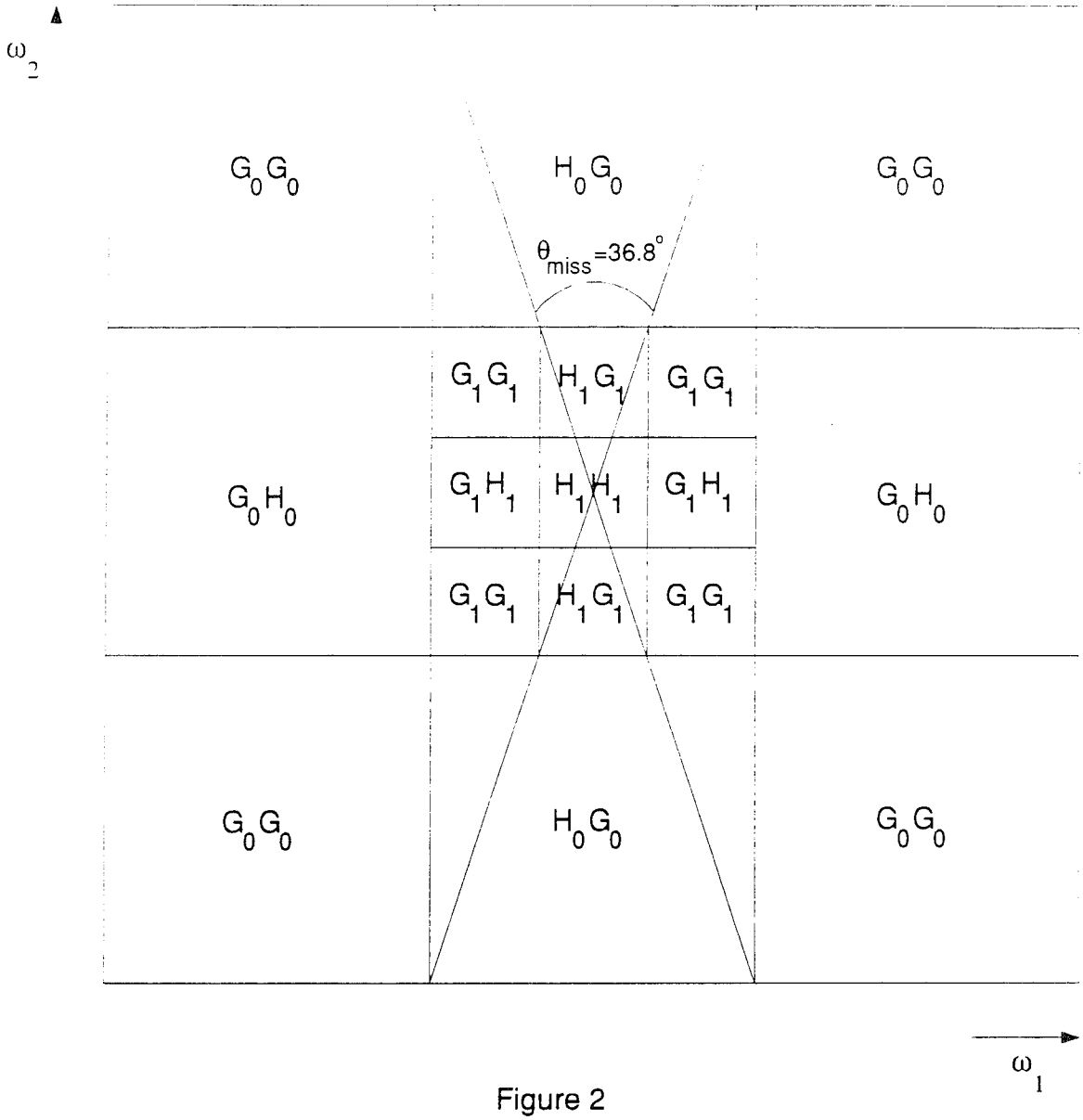

ω2
G0G0
H0G0
G0G0
θmiss=36.8°
G1G1
H1G1
G1G1
G0H0
G1H1
H1H1
G1H1
G0H0
G1G1
H1G1
G1G1
G0G0
H0G0
G0G0
ω1

Figure 2

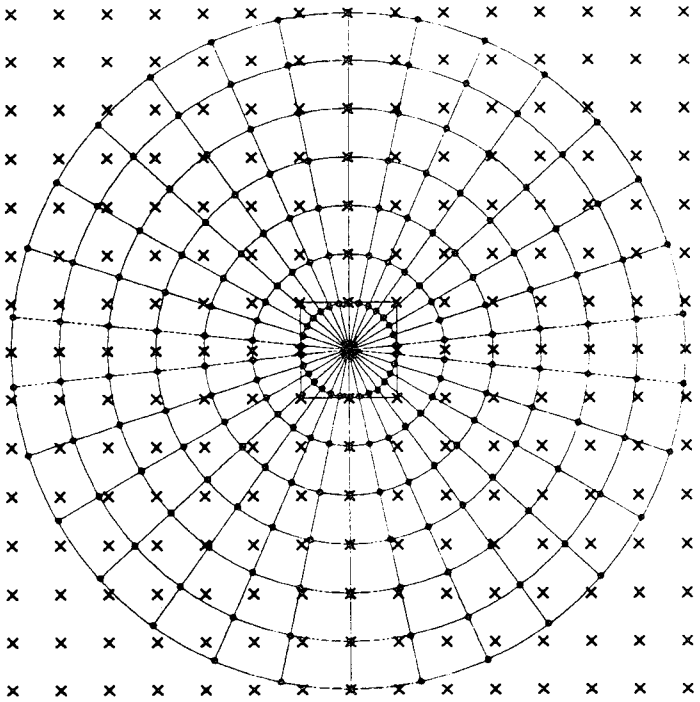


Figure 3

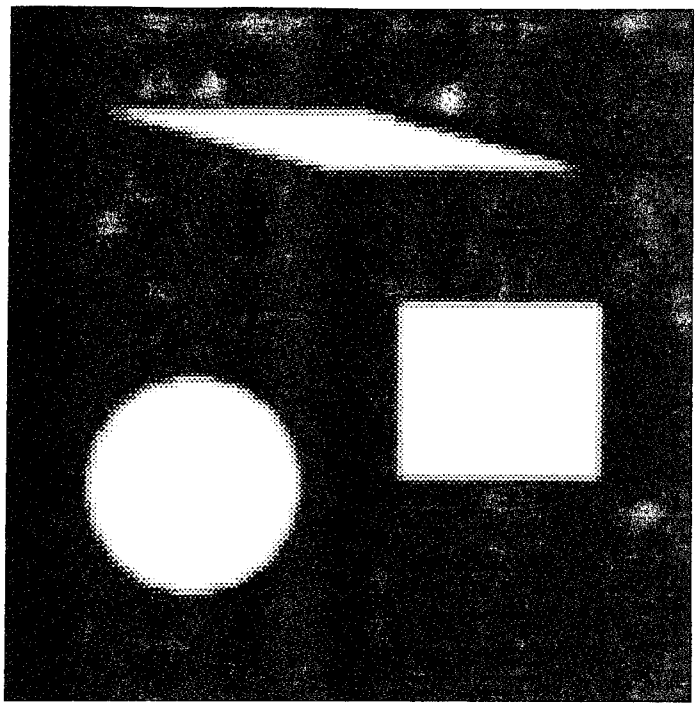


Figure 4.

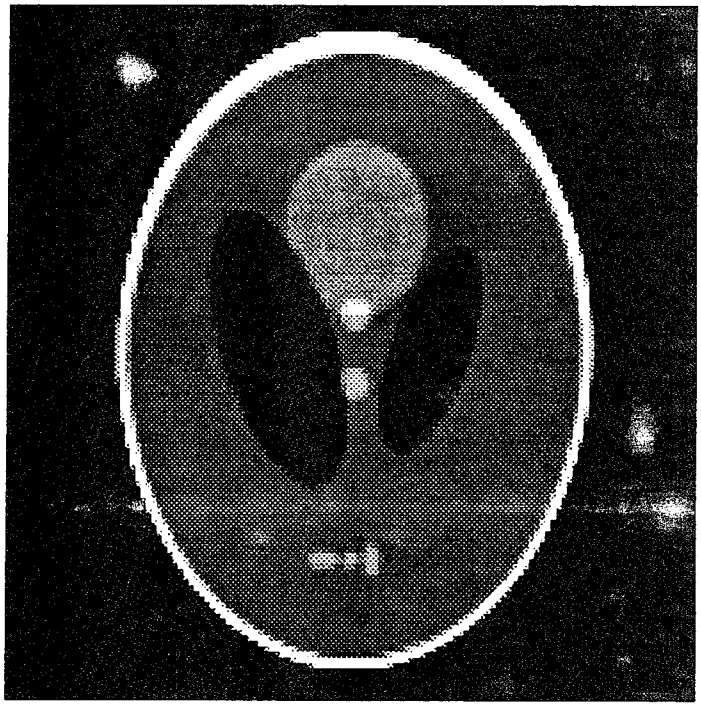


Figure 5.

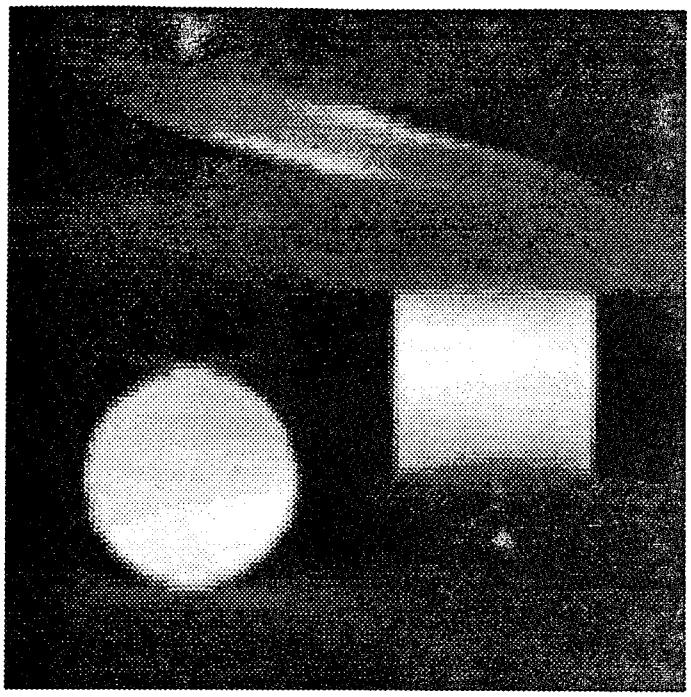


Figure 6.

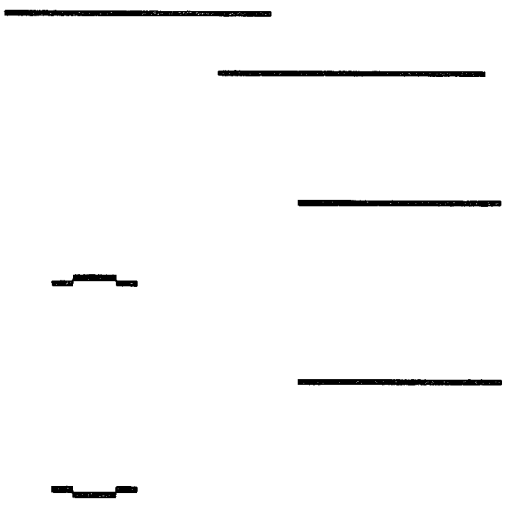


Figure 7.

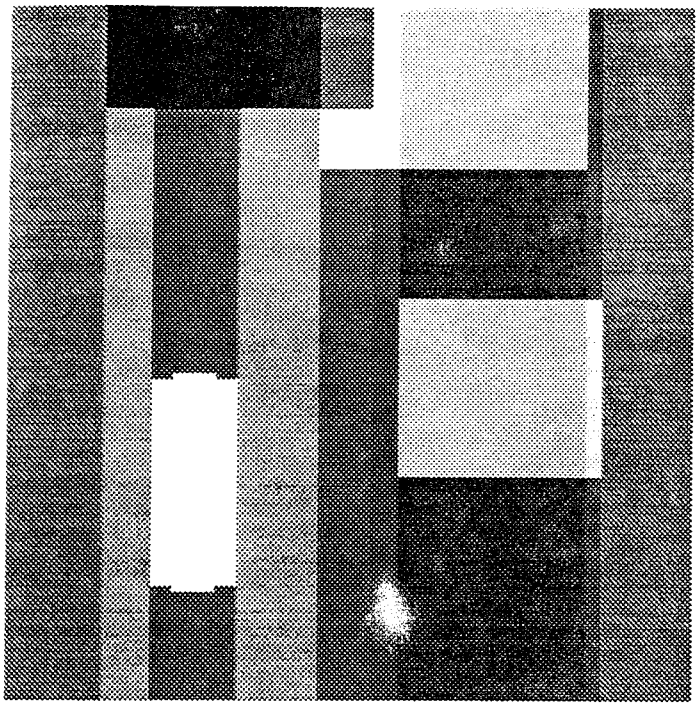


Figure 8.

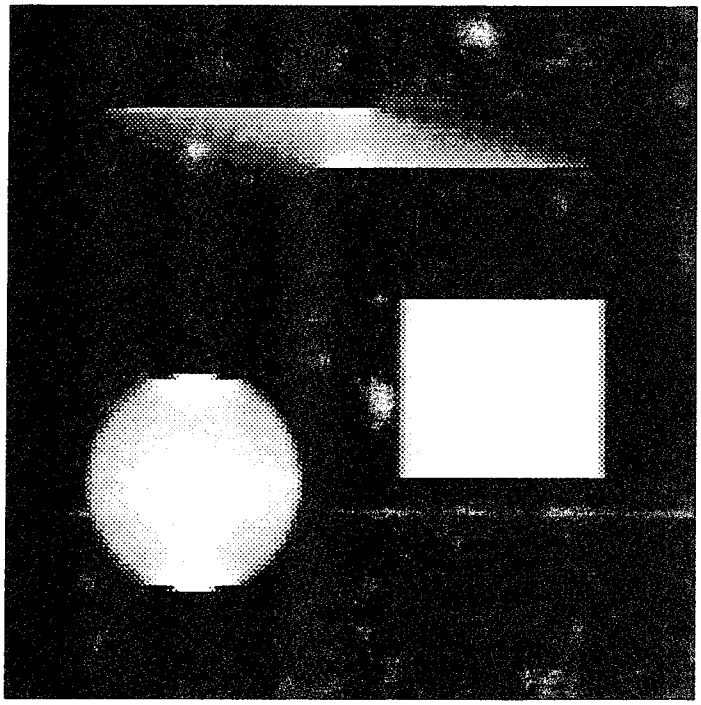


Figure 9.

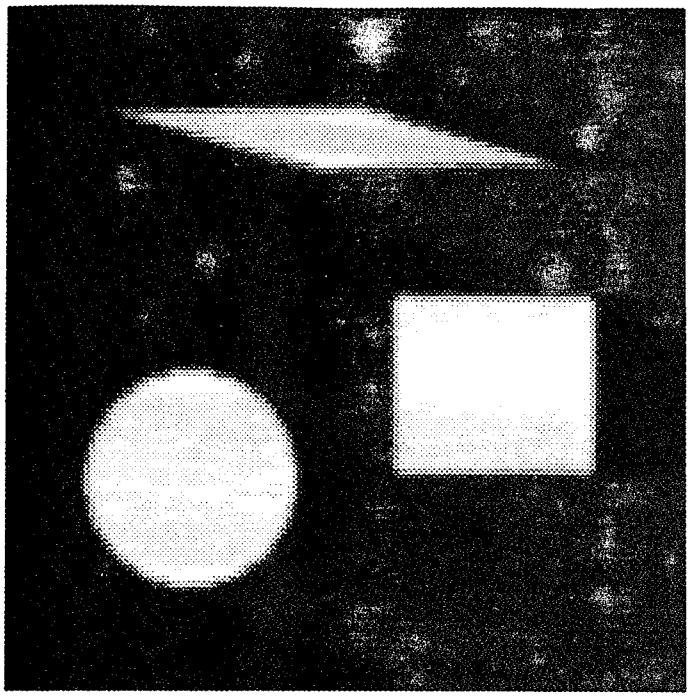


Figure 10.

Figure 11.

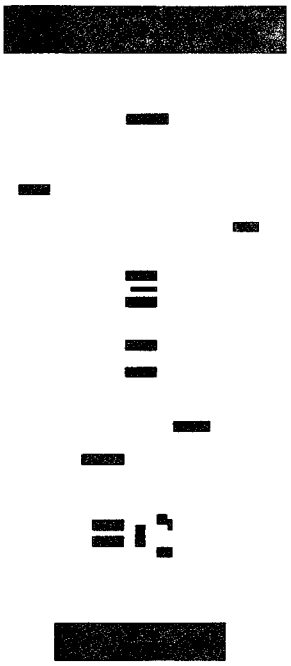


Figure 12.

Figure 13.

APPENDIX N

B. Sahiner and A.E. Yagle, "Local Reconstruction from Projections using Exponential Radial Sampling," submitted to IEEE Trans. Image Proc., July 1993.

This paper shows that the local tomography problem of reconstructing only a small region of interest (ROI) from a limited set of projections can be solved by sampling the projections at a rate that decreases exponentially with distance from the ROI. This reconstructs the ROI with high resolution, and the remainder of the image at lower resolution. The algorithm is also much faster than conventional filtered backprojection.

LOCAL RECONSTRUCTION FROM PROJECTIONS USING EXPONENTIAL RADIAL SAMPLING

Berkman Sahiner and Andrew E. Yagle

Dept. of Electrical Engineering and Computer Science

The University of Michigan, Ann Arbor, Michigan 48109-2122

June 1993

Abstract

We combine several ideas, including nonuniform sampling and circular harmonic expansions, into a new procedure for reconstructing a small region of interest (ROI) of an image from a set of its projections that are densely sampled in the ROI and coarsely sampled outside the ROI. Specifically, the radial sampling density of both the projections and the reconstructed image decreases exponentially with increasing distance from the ROI. The problem and data are reminiscent of the recently-formulated local tomography problem; however, our algorithm reconstructs the ROI of the image itself, not the filtered version of it obtained using local tomography. The new algorithm has the added advantages of speed (it can be implemented entirely using the FFT) and parallelizability (each image harmonic is computed independently). Numerical examples compare the new algorithm to filtered backprojection.

EDICS Number: 6.3.1. Fax:313-763-1503 email:aey@eecs.umich.edu

I. INTRODUCTION

The problem of image reconstruction from a complete set of projections is to compute an image $\mu(x,y)$ from a complete set of its line integrals $p(r,\theta)$, defined as

$$p(r,\theta) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mu(x,y)\delta(r - xcos\theta - y\sin\theta)dxdy. \quad (1)$$

The most common procedure for reconstruction from a complete set of projections is filtered backprojection (FBP). In FBP the projections $p(r,\theta)$ are first filtered with a filter $h(r)$ whose Fourier transform $\hat{h}(w) \approx |w|$ up to some cutoff frequency, and is windowed to zero for higher frequencies. These filtered projections are then backprojected [1]. When the projections are sampled in the angular and radial variables, but cover the entire extent of the image, FBP still yields quite satisfactory results. The resolution of the reconstructed image is determined by the sampling densities in r and θ of $p(r,\theta)$ and the cutoff frequency of $\hat{h}(w)$.

In many applications, it is not possible to obtain a complete set of projections which are sampled densely enough to attain the desired resolution over the entire support of $\mu(x,y)$. For example, X-ray dose limitations, or time constraints when imaging a moving object [2], may preclude such a large number of projections. If the entire support of $\mu(x,y)$ is covered, but projections are not sampled densely enough, then the desired resolution is not attained. If the projections are dense enough around some region of interest (ROI) of $\mu(x,y)$, but do not cover the entire support of $\mu(x,y)$, then a good reconstruction using FBP is not possible, due to the infinite support of the filter $h(r)$ (ideally, $h(r)$ is a derivative-Hilbert transform $\mathcal{H}d/dr$).

This is one reason why local tomography was introduced in [2]-[4]. In local tomography, $h(r)$ no longer approximates $\mathcal{H}d/dr$. Instead, the local filter $h(r) = d^2/dr^2 + \alpha\delta(r)$ is used, and the Fourier transform $\hat{\tilde{\mu}}(w_1,w_2)$ of the reconstructed image $\tilde{\mu}(x,y)$ is related to the Fourier transform $\hat{\mu}(w_1,w_2)$ of $\mu(x,y)$ by $\hat{\tilde{\mu}}(w_1,w_2) = \sqrt{w_1^2+w_2^2}\hat{\mu}(w_1,w_2) + \alpha\hat{\mu}(w_1,w_2)/\sqrt{w_1^2+w_2^2}$. The idea is that since this $h(r)$ is local, only projections passing through the ROI are used; no other projections need be taken. Local tomography was used successfully to image the coronary arterial tree in [2]. However, it is clear that $\tilde{\mu}(x,y) \neq \mu(x,y)$; for example, constant regions

of $\mu(x,y)$ tend to become cup-shaped functions $\tilde{\mu}(x,y)$ [4]. Furthermore, local tomography is even more susceptible to noise than FBP, due to the extra noise-amplifying $\sqrt{w_1^2+w_2^2}$ filter.

In this paper, we introduce a different type of local tomography, based on exponential radial sampling of the image and projections. We assume that we are interested in obtaining high resolution only in a small ROI; outside this region, high resolution is not very important. Without loss of generality, we assume that the ROI is centered on the origin (this can easily be achieved by translating $\mu(x,y)$). The angular sampling is conventional equiangular sampling, i.e., $p(r,\theta)$ is sampled in θ at angles $\theta_n = \frac{2\pi}{N}n$, $n = 0, 1, \ldots, N-1$. However, the radial sampling in r in $p(r,\theta)$ and ρ in $\mu(x,y) = \mu(\rho,\phi)$ (polar coordinates) is exponential, i.e., $p(r,\theta)$ is sampled in r at distances $r_k = r_1 e^{(k-1)\Delta}, k \geq 1$.

This means that the samples are very dense around the origin (i.e., in the ROI), and the sampling density decreases exponentially with increasing distance from the origin. This gives us good resolution around the origin (in the ROI), and poor resolution far away from the origin (which is irrelevant). These remarks apply both to the data (the projections $p(r,\theta)$) and the reconstructed image $\mu(\rho,\phi)$. Although the exponential decrease of sampling density with increasing r is not as sharp as the abrupt drop of sampling density to zero in local tomography of [2]-[4], it is quite steep, regardless of the value of Δ, and it is clearly in the spirit of localizing the projection data in a ROI. It shares the advantages of the local tomography of [2]-[4] (viz., using less data, with attendant smaller X-ray exposure). And it has a significant advantage over the local tomography of [2]-[4]: $\mu(x,y)$, not $\tilde{\mu}(x,y)$, is computed in the ROI.

For image reconstruction using exponential radial sampling, we use the circular harmonic decomposition, which is a Fourier expansion in the angular variable θ or ϕ. This decomposition has been applied to reconstruction from projections in [5]-[7]. However, [5]-[7] used either continuous variables or uniform sampling, while we use exponential radial sampling. This creates two advantages: (1) it results in a local tomography problem, as described above; and (2) the reconstruction formula can be written as a regular convolution for each harmonic. Since the fast Fourier transform (FFT) can be used to implement these convolutions, all in parallel,

this results in a reconstruction algorithm that is an order of magnitude faster than FBP. In [9] exponential sampling was used to make the Abel transform a convolution; however, that was a one-dimensional problem, while this paper treats a two-dimensional local tomography problem.

We use a bilinear polar-to-rectangular coordinate conversion algorithm to display the final image. The conversion algorithm can produce images which represent $\mu(x,y)$ at different resolutions, i.e., a small area of $\mu(x,y)$ in the ROI with very good resolution, or a larger area of $\mu(x,y)$ with less resolution. Since resolution increases exponentially with decreasing distance from the origin, we will be able to zoom in on the ROI with little loss of resolution.

The paper is organized as follows. In Section II, we give definitions and review circular harmonic image reconstruction algorithms of [5]-[7]. In Section III, we apply exponential sampling and derive a new fast algorithm for reconstructing a small ROI from exponential radial sampled projection data. In Section IV, we present and discuss some numerical examples, and compare our results with those from FBP with regular sampling. Section V concludes with a summary.

II. CIRCULAR HARMONIC IMAGE RECONSTRUCTION

Let $\mu(\rho,\phi)$ denote the image in polar coordinates. Since both the image and its projection $p(r,\theta)$ are periodic in the angular variable with period 2π, they can be expanded in Fourier series (circular harmonic decompositions [5]-[7])

$$\mu(\rho,\phi) = \sum_{n=-\infty}^{\infty} \mu_n(\rho) e^{jn\phi}; \qquad p(r,\theta) = \sum_{n=-\infty}^{\infty} p_n(r) e^{jn\theta}, \tag{2}$$

where

$$\mu_n(\rho) = \frac{1}{2\pi}\int_0^{2\pi} \mu(\rho,\phi) e^{-jn\phi} d\phi; \qquad p_n(r) = \frac{1}{2\pi}\int_0^{2\pi} p(r,\theta) e^{-jn\theta} d\theta \tag{3}$$

are the circular harmonics of $\mu(\rho,\phi)$ and $p(r,\theta)$, respectively. Since $\mu(\rho,\phi)$ and $p(r,\theta)$ are both real, we have $\mu_{-n}(\rho) = \mu_n^*(\rho)$ and $p_{-n}(r) = p_n^*(r)$. We now show that $\mu_n(\rho)$ can be computed from $p_n(r)$, independently for each n.

A. *Original Cormack Formula*

Cormack [5] was the first to use circular harmonics in reconstruction from projections. He showed that the circular harmonics $\mu_n(\rho)$ of the image can be obtained from the circular harmonics $p_n(r)$ of the projections using

$$\mu_n(\rho) = -\frac{1}{\pi}\int_\rho^\infty \frac{p_n'(r)T_n(r/\rho)}{\sqrt{r^2-\rho^2}}dr, \tag{4}$$

where $p_n'(r) = dp_n(r)/dr$ and $T_n(x) = \cos(n\cos^{-1}x)$ is the Chebyshev polynomial of the first kind of order n.

The Cormack formula (4) has been called the "causal, unstable" form of circular harmonic image reconstruction [6]. Equation (4) is causal in that $\mu_n(\rho)$ depends only on $p_n(r)$ for $r \geq \rho$; thus (4) solves the exterior Radon transform problem of reconstructing $\{\mu(\rho,\phi), \rho > R_0\}$ from $\{p(r,\theta), r > R_0\}$. It is unstable in that for large r/ρ, $T_n(r/\rho)$ behaves like $(r/\rho)^n$, therefore the integrand in (4) becomes very large, especially for small ρ. This instability makes (4) almost useless for image reconstruction from projections. We do not use (4) in this paper.

B. *Stable Noncausal Formula*

There also exists a "noncausal, stable" form of circular harmonic reconstruction [6], [7]:

$$\mu_n(\rho) = \frac{1}{\pi}\int_0^\rho U_{n-1}(r/\rho)p_n'(r)dr - \frac{1}{\pi}\int_\rho^\infty \frac{\exp(-n\cosh^{-1}(r/\rho))}{\sqrt{r^2-\rho^2}}p_n'(r)dr, \tag{5}$$

where $U_n(x) = \sin((n+1)\cos^{-1}x)/\sin(\cos^{-1}x)$ is the Chebyshev polynomial of the second kind of order n. Equation (5) is noncausal in that $\mu_n(\rho)$ depends on $p_n(r)$ for all r, not just $r \geq \rho$, but the integrands are stable, in that they are bounded as $n \to \infty$. Chapman and Cary [7] have used the noncausal, stable form (5) to perform fairly accurate reconstructions from regularly sampled projections.

C. *Reformulation of (4) as Convolution*

We now apply an idea first used in [9] on the Abel transform to the Cormack formula (4). Although we do not use the result further, it provides a simple illustration of the more complicated

transformation we will apply in Section III to the discretized form of (5).

Make the change of variables $r = e^{-t}$, $\rho = e^{-\tau}$, and define $\bar{\mu}_n(\tau) = \mu_n(\rho = e^{-\tau})$ and $\bar{p}_n(t) = p_n(r = e^{-t})$. Then (4) becomes

$$\begin{aligned} \bar{\mu}_n(\tau) &= \frac{1}{\pi}\int_{-\infty}^{\infty}\left[\frac{d\bar{p}(t)}{dt}\right] e^t \frac{T_n(e^{\tau-t})}{\sqrt{1-e^{-2(\tau-t)}}} 1(\tau-t) \\ &= \frac{1}{\pi}\bar{p}_n'(t)e^t * \frac{T_n(e^t)}{\sqrt{1-e^{-2t}}} 1(t), \end{aligned} \tag{6}$$

where $*$ denotes convolution and $1(t)$ is the unit step function. Thus the transform defined in (4) has become a simple filtering operation, which can be implemented using the FFT. Indeed, it suggests that the exponential warping of r and ρ is a more natural formulation, since (4) represents a time-varying filter and (6) represents a time-invariant filter.

III. DERIVATION OF THE ALGORITHM

A. *Problem Specification*

The problem that we solve in this section is defined as follows. Given samples of the projections of an image, where the sampling density is exponential in the radial variable and equiangular in the angular variable, compute the image on the same grid. That is, given

$$\left\{p(r_k,\theta_n),\ r_k = r_1 e^{(k-1)\Delta},\ k = 1,\ldots,K; \quad \theta_n = \frac{2\pi}{N}n,\ n = 0,1,\ldots,N-1\right\} \tag{7}$$

compute $\{\mu(\rho_k,\phi_n)\}$ for analogous values of ρ_k and ϕ_n. For convenience we define $r_0 = \rho_0 = 0$.

We assume that: (1) the image (and hence the projections also) is known to have its support inside a disk of radius R; and (2) the image (and hence the projections also) is known to have only N circular harmonics $\mu_n(\rho)$ significantly different from zero. $r_1 = \rho_1$ is the smallest radius of interest, and $r_K = \rho_K = R$, so that $(K-1)\Delta = \ln(R/r_1)$. Note that the grid is very dense around r_1, and much sparser around r_K, dropping off exponentially with increasing radius, regardless of the size of Δ.

B. *Discretization*

We first discuss how to obtain the harmonics $\mu_n(\rho)$ of the image from the harmonics $p_n(r)$ of the projections. Since $\mu_{-n}(\rho) = \mu_n^*(\rho)$, we require image harmonics only for $n \geq 0$. Changing variables from r to $x = \cos^{-1}(r/\rho)$ in the first integral of (5) and $x = \cosh^{-1}(r/\rho)$ in the second integral of (5), and substituting $\rho = \rho_j$, (5) can be written as [7]

$$\begin{aligned}\mu_n(\rho_j) &= \frac{1}{\pi}\int_0^{\pi/2} \sin(nx) p_n'(\rho_j \cos x)dx - \frac{1}{\pi}\int_0^{\cosh^{-1}(R/\rho_j)} e^{-nx} p_n'(\rho_j \cosh x)dx \\ &= \frac{1}{\pi n}\int_{x=0}^{x=\pi/2} p_n'(\rho_j \cos x)d(\cos nx) + \frac{1}{\pi n}\int_{x=0}^{x=\cosh^{-1}(R/\rho_j)} p_n'(\rho_j \cosh x)d(e^{-nx}). \quad (8)\end{aligned}$$

We now generalize the result of [7]. Let $\{x_{j,k}\}_{k=0}^K$ be any real numbers such that

$$0 = x_{j,j} \leq x_{j,j-1} \leq \ldots \leq x_{j,0} = \pi/2 \text{ and } 0 = x_{j,j} \leq x_{j,j+1} \leq \ldots \leq x_{j,K} = \cosh^{-1}(R/\rho_j). \quad (9)$$

Then, following [7], it is easily shown that a first approximation to (8) for $n \neq 0$ is given by

$$\mu_n(\rho_j) = \frac{1}{\pi n}\sum_{k=0}^{j-1} a_{n,j}(k)\left(\cos(nx_{j,k+1}) - \cos(nx_{j,k})\right) + \frac{1}{\pi n}\sum_{k=j}^{K-1} a_{n,j}(k)\left(e^{-nx_{j,k+1}} - e^{-nx_{j,k}}\right) \quad (10)$$

where $a_{n,j}(k)$, which approximates $p_n'(\rho_j \cos x)$ or $p_n'(\rho_j \cosh x)$, is defined as

$$a_{n,j}(k) = \begin{cases} \dfrac{p_n(\rho_j \cos x_{j,k+1}) - p_n(\rho_j \cos x_{j,k})}{\rho_j \cos x_{j,k+1} - \rho_j \cos x_{j,k}}, & k = 0, \ldots, j-1 \\ \\ \dfrac{p_n(\rho_j \cosh x_{j,k+1}) - p_n(\rho_j \cosh x_{j,k})}{\rho_j \cosh x_{j,k+1} - \rho_j \cosh x_{j,k}}, & k = j, \ldots, K-1. \end{cases} \quad (11)$$

C. *Exponential Radial Sampling*

All of the following results are new. Instead of choosing $x_{j,k} = \cos^{-1}(k/j)$ or $\cosh^{-1}(k/j)$ as in [7], we choose

$$x_{j,k} = x_{j-k} = \begin{cases} \cos^{-1}\left(e^{\Delta(j-k)}\right), & 1 \leq k \leq j \\ \cosh^{-1}\left(e^{\Delta(j-k)}\right), & j \leq k \leq K, \end{cases} \quad (12)$$

and we recall from (9) that $x_{j,0} = \pi/2$ for all j. This results in

$$a_{n,j}(k) = a_n(k) = \frac{p_n(r_{k+1}) - p_n(r_k)}{r_{k+1} - r_k}, \quad (13)$$

which again is clearly a discrete representation of $p'_n(r)$. Defining

$$s_n(j) = \begin{cases} \cos nx_{j-1} - \cos nx_j & j > 0 \\ e^{-nx_{j-1}} - e^{-nx_j} & j \leq 0 \end{cases} \tag{14}$$

and substituting in (10), we obtain the main result of this paper:

$$\mu_n(\rho_j) = \frac{1}{n\pi}\left[a_n(0)(\cos nx_{j-1} - \cos n\pi/2) + \sum_{k=1}^{K-1} a_n(k)s_n(j-k)\right], n \neq 0, j \neq 0. \tag{15}$$

Equation (15) computes the exponentially-sampled image harmonics $\mu_n(\rho_j)$ from the exponentially-sampled projection harmonics $p_n(r_k)$. First, (13) "differentiates" $p_n(r_k)$; then the result is convolved with $s_n(j)$ to compute $\mu_n(\rho_j)$ (the first term in (15) is an end effect). Note that this is reminiscent of (6); however, it comes from the noncausal, stable form (5), not the causal, unstable Cormack formula (4).

Note that (15) cannot be obtained by a simple discretization of the continuous result of [6], due to the end effect and sampling points. In [6] the change of variables $r = e^{-t}$ was applied directly to (5) without much discussion of the result, and no discretization was performed. Our result (15) differs from the results of [6] and [7] in the following ways: (1) unlike [6], we perform the change of variables (8); (2) unlike [6], our result is explicitly discrete and directly suitable for computer processing of sampled data; and (3) unlike [7], we use exponentially-sampled data. These differences, and the novel application to local tomography, justify calling (15) a new result.

For the special cases $n = 0$ and $j = 0$, it can be shown that [7]

$$\mu_0(\rho_j) = -\frac{1}{\pi}\sum_{k=j}^{K-1} a_n(k)(x_{j-k-1} - x_{j-k}) \tag{16}$$

and

$$\mu_n(\rho_0) = \mu_n(0) = \begin{cases} -\frac{1}{\pi}\left[2a_0(0) + \Delta\sum_{k=1}^{K-1} a_0(k)\right], & n = 0 \\ 0, & n \neq 0. \end{cases} \tag{17}$$

D. *Equiangular Sampling*

Since we are given $p(r_k, \theta_n)$ for $\theta_n = \frac{2\pi}{N}n, n = 0, \ldots, N-1$, we consider $p(r, \theta) = 0$ for $\theta \neq \frac{2\pi}{N}n$. Since $p(r_k, \theta)$ is discrete and periodic in θ, its Fourier transform is also discrete and

periodic. Since by assumption $p(r_k, \theta)$ is angularly bandlimited, (3) becomes the discrete Fourier transform

$$p_n(r_k) = \frac{1}{2\pi} \sum_{l=0}^{N-1} p(r_k, \theta_l) e^{-j\frac{2\pi}{N}nl}, \quad n = -N/2+1, \ldots, N/2. \tag{18}$$

Similarly, (2) also becomes a discrete Fourier transform.

In actual application, $\mu(\rho, \phi)$ (and hence $p(r, \theta)$ also) is unlikely to be bandlimited to N harmonics. Hence some aliasing may be expected. However, the higher-order harmonics tend to be smaller near the origin, i.e., in the ROI. Also, to reduce ringing effects caused by the sudden truncation of the circular harmonic expansion (2), we use not (2) but a windowed version of (2)

$$\mu(\rho_j, \phi_l) = \sum_{n=-N/2+1}^{N/2} w_n \mu_n(\rho_j) e^{j\frac{2\pi}{N}nl}, \quad l = 0, \ldots, N-1, \tag{19}$$

where w_n implements a Hamming window. A similar idea was used in [8].

E. Summary of Algorithm

Given $\left\{p(r_k, \theta_n),\ r_k = r_1 e^{(k-1)\Delta},\ k = 1, \ldots, K; \quad \theta_n = \frac{2\pi}{N}n,\ n = 0, 1, \ldots, N-1\right\}$:

1. Compute $p_n(r_k)$ using (18) and FFT, in parallel in n;
2. Compute $a_n(k)$ using (13), in parallel in n;
3. Compute $\mu_n(\rho_j)$ using (15) and FFT, in parallel in n;
4. Compute $\mu(\rho_j, \phi_l)$ using (19) and FFT, in parallel in n.

Finally, we compute the number of operations required to carry out our algorithm. All of the equations can be implemented using the FFT; for $N = K$, this requires $O(N^2 \log N)$ operations. Since both (13) and (15) can be parallelized in n, an even greater computational speedup is possible. By comparison, FBP requires $O(N^3)$ operations to compute the image on a $N \times N$ grid. The computational savings is thus a factor of $O(N/\log N)$.

IV. NUMERICAL EXAMPLES

A. *Numerical Procedures*

To demonstrate the effectiveness of our algorithm in achieving high resolution in a region of interest of the image while minimizing artifacts, we present some simulations using the Shepp-Logan phantom shown in Figure 1. The ROI is defined to consist of the three small ovals at the bottom. Accordingly, the image has been translated in the y-direction by 0.605, so that the small circle, surrounded by two small ovals, is now in the center of the image. All of the images shown in this section are displayed on a 256×256 grid, and for the projections $N = 512$ and $K = 128$ (i.e., 128 views at 512 angles). For exponential sampling, $r_1 = 0.01$, $r_K = R = 1.6$, so $\Delta = \ln(1.6/0.01)/127 = 0.039962$.

To display our reconstructed images, we use a bilinear polar-to-rectangular interpolation algorithm. That is, let (x, y) be a point at which interpolation is desired. Let $(x, y) = (\rho, \phi)$ lie inside the trapezoid with corners having polar coordinates $\{(\rho_k, \phi_k), (\rho_{k+1}, \phi_k), (\rho_k, \phi_{k+1}), (\rho_{k+1}, \phi_{k+1})\}$. Then the interpolated value is

$$\begin{aligned} \mu(x,y) &= \frac{ND}{2\pi}[(\rho-\rho_k)(\phi-\phi_k)\mu(\rho_{k+1},\phi_{k+1}) + (\rho-\rho_k)(\phi_{k+1}-\phi)\mu(\rho_{k+1},\phi_k) \\ &+ (\rho_{k+1}-\rho)(\phi-\phi_k)\mu(\rho_k,\phi_{k+1}) + (\rho_{k+1}-\rho)(\phi_{k+1}-\phi)\mu(\rho_k,\phi_k)] \end{aligned} \tag{20}$$

where $\frac{N}{2\pi} = \frac{1}{\phi_{k+1}-\phi_k}$ and $D = \frac{1}{\rho_{k+1}-\rho_k}$.

Suppose that we are using this interpolation algorithm to display a 256×256 image covering the entire phantom. Then pixels far away from the origin of the image will be interpolated using reconstructed polar values $\mu(\rho_j, \phi_n)$ that are not very close to those pixels, while pixels close to the origin of the image will be interpolated using $\mu(\rho_j, \phi_n)$ that are very close to them. In such a situation, many values of $\mu(\rho_j, \phi_n)$ with very small ρ_j will not be used at all in the interpolation to a rectangular grid, since they will not be closest to any pixel. Hence, we may "zoom in" to the origin of the image, i.e., display 256×256 images that cover smaller and smaller areas around the origin.

To evaluate our results, we compare them to analogous results obtained using the same number (NK) of samples, but with uniform sampling in r, reconstructed using FBP [1]. The projections $p(r_k, \theta_n)$ are collected for $r_k = \frac{Rk}{K} \times \frac{1}{A}, \quad k = 0, \ldots, K-1$; for $r \geq r_K$, we set $p(r, \theta_n) = p(r_{K-1}, \theta_n)$ (this proved to be surprisingly effective, much more effective than setting these values to zero). The parameter A determines the maximum radius that is sampled; as A increases, a smaller region is sampled more finely.

B. Discussion of Results

Figs. 2-5 compare the results from our algorithm (Figs. 2a-5a) to those from FBP [1] (Figs. 2b-5b). The successive close-up views were obtained as follows. For our algorithm, the excess of $\mu(\rho_k, \theta_n)$ near the origin allows us to "zoom in" to the origin, as explained above. For FBP, we used $A = 1$, 2, 4 and 8 to generate Figs. 2b-5b, respectively, as explained above.

Comparing Fig. 2a and Fig. 2b, it is clear that the overall FBP image is sharper, but in the ROI our exponential sampling algorithm produces a sharper image. In Fig. 2a, note the poor resolution at the top of the image. This is as expected – this region is far from the origin, so its resolution should be poor. But the ROI at the bottom of the image is very sharp.

Zooming in on the ROI in Figs. 3-5, our algorithm continues to produce a sharp image, with only a few faint circular artifacts. In contrast, FBP produces an image in which the three ovals are almost washed out. This is the familiar "dishing" artifact (named by Kak [10]), in which the image is artificially bright near its center. Note that this is a very serious error, since the three-oval ROI lies inside another oval, which must also be reconstructed correctly (i.e., the constant but non-zero background must also be reconstructed).

The dishing artifact in FBP is caused by the infinite support of the filter $h(r)$. In Fig. 5c, $A = 1$ and FBP is used to reconstruct only the square $[-0.2,0.2] \times [-0.2,0.2]$. Since the projections now cover the support of the image, there are no dish artifacts, but the reconstruction is much more blurred than the result of our algorithm (Fig. 5a). Also note that FBP requires a different set of projections for each of Figs. 2b-5b, while our algorithm uses the same set of

projections for each of Figs. 2a-5a.

V. CONCLUSION

We have proposed a new algorithm for reconstructing a small ROI of an image from a set of its projections that have been exponentially sampled in the radial variable. Unlike the similar local tomography problem, the ROI of the image itself, not a filtered version of it, is reconstructed. The algorithm draws on previous work on using the circular harmonic expansion for image reconstruction from projection. The major contributions of this paper include: (1) recognition of the applicability of an exponential radial sampling density to a local tomography problem; (2) implementation of a discrete equation using exponential sampling (previous work used a continuous exponential transform in a continuous equation); and (3) numerical examples demonstrating the better performance of the algorithm on local tomography, as compared to FBP. The algorithm is also very fast (implemented entirely using the FFT) and parallelizable (each harmonic is treated independently).

An important topic for further research is to determine the minimum number of view angles required to obtain a good reconstruction in the ROI. Fewer image harmonics should be needed near the origin, and since the harmonics are independent, fewer projection harmonics are needed. The problem is to determine a small set of view angles that would enable computation of only the smallest projection harmonics. This is a topic of current research.

ACKNOWLEDGMENT

This work was supported by the Office of Naval Research under grant #N00014-90-J-1897.

REFERENCES

[1] L. A. Shepp and B. F. Logan, "The Fourier reconstruction of a head section," IEEE Trans. Nucl. Sci., vol. NS-21, pp. 21-42, 1974.

[2] W. J. T. Spyra, A. Faridani, K. T. Smith and E. L. Ritman, "Computed tomographic imaging of the coronary arterial tree – Use of local tomography," IEEE Trans. Med. Imaging, vol. MI-9, pp. 1-4, 1990.

[3] A. Faridani, F. Keinert, F. Natterer, E. L. Ritman and K. T. Smith, "Local and global tomography," in *Signal Processing* (IMA Volumes in Mathematics and its Applications), New-York: Springer-Verlag, 1990.

[4] A. Faridani, E. L. Ritman, K. T. Smith, "Local tomography," SIAM J. Appl. Math., vol. 52, pp. 459-484, 1992.

[5] A. M. Cormack, "Representation of a function by its line integrals, with some radiological applications," J. Appl. Physics, vol. 43, pp. 2722-2727, 1963.

[6] E. W. Hansen, "Theory of circular harmonic image reconstruction," J. Opt. Soc. Am., vol. 71, pp. 304-308, 1981.

[7] C. H. Chapman and P. W. Cary, "The circular harmonic Radon transform," Inverse Problems, vol. 2, pp. 23-44, 1986.

[8] W. E. Higgins and D. C. Munson, "A Hankel transform approach to tomographic image reconstruction," IEEE Trans. Med. Imaging, vol. MI-7, pp. 59-72, 1988.

[9] E. W. Hansen and P.-L. Law, "Recursive methods for computing the Abel transform and its inverse," J. Opt. Soc. Am. A, vol.2, pp. 510-520, 1985.

[10] A. C. Kak, M. Slaney, *Principles of Computerized Tomographic Imaging*, New York: IEEE Press, 1987.

Figure Headings

Fig. 1. The Shepp-Logan phantom.

Fig. 2a. The result from our algorithm, the entire image in the square $[-1.6,1.6] \times [-1.6,1.6]$ is displayed on a 256×256 grid.

Fig. 2b. The result from FBP, with sampling parameter $A = 1$.

Fig. 3a. Same as Fig. 2a., only the square $[-0.8,0.8] \times [-0.8,0.8]$ is displayed.

Fig. 3b. The result from FBP, with sampling parameter $A = 2$.

Fig. 4a. Same as Fig. 2a., only the square $[-0.4,0.4] \times [-0.4,0.4]$ is displayed.

Fig. 4b. The result from FBP, with sampling parameter $A = 4$.

Fig. 5a. Same as Fig. 2a., only the square $[-0.2,0.2] \times [-0.2,0.2]$ is displayed.

Fig. 5b. The result from FBP, with sampling parameter $A = 8$.

Fig. 5c. The result from FBP with sampling parameter $A = 1$, only the square $[-0.2,0.2] \times [-0.2,0.2]$ is displayed.

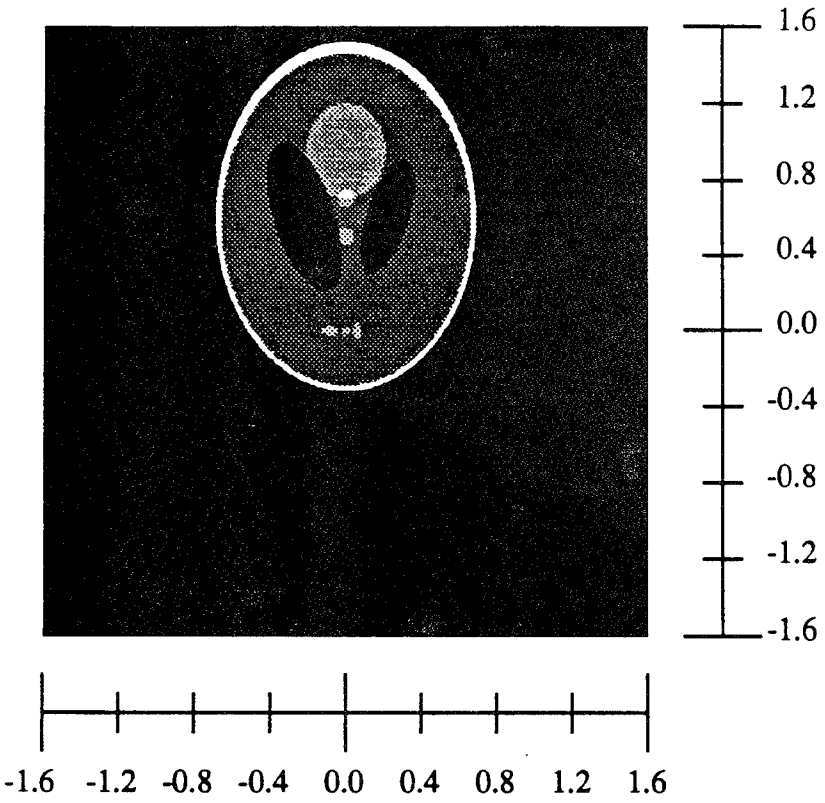


Figure 1.

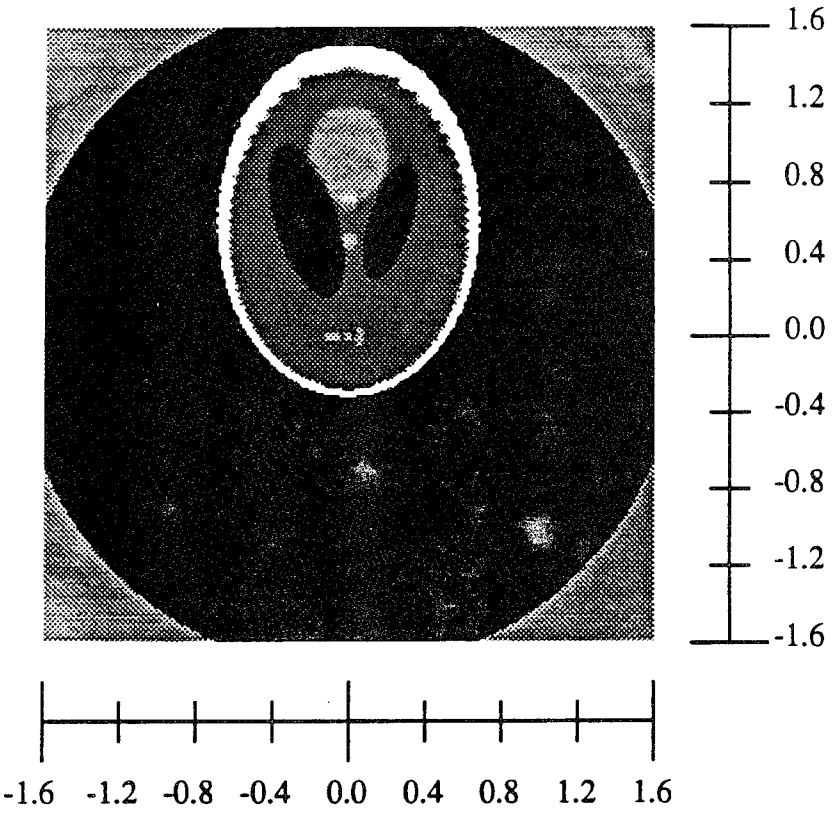


Figure 2a.

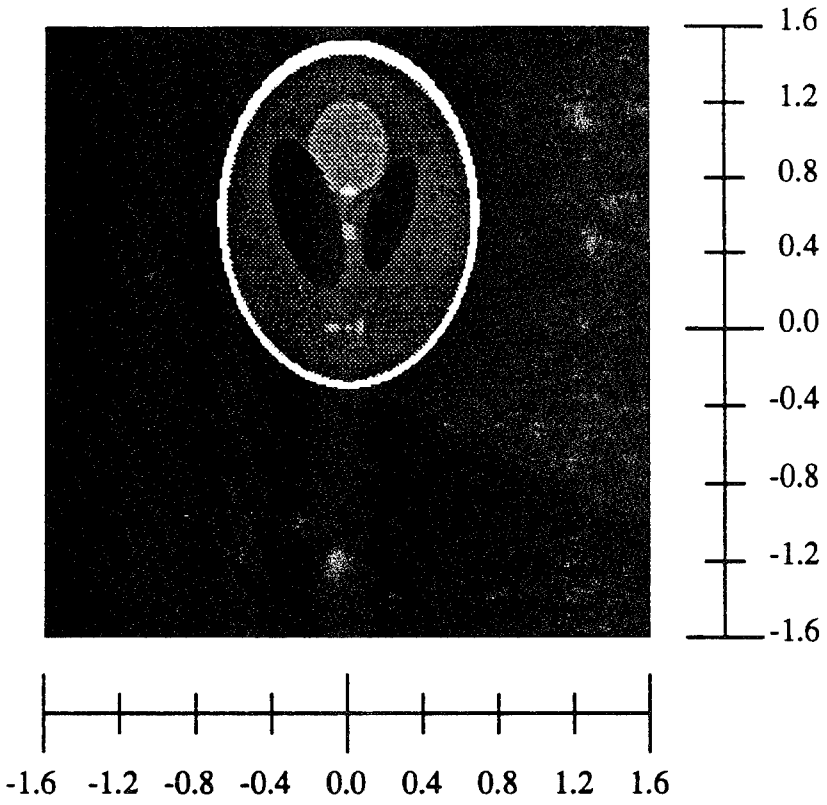


Figure 2b.

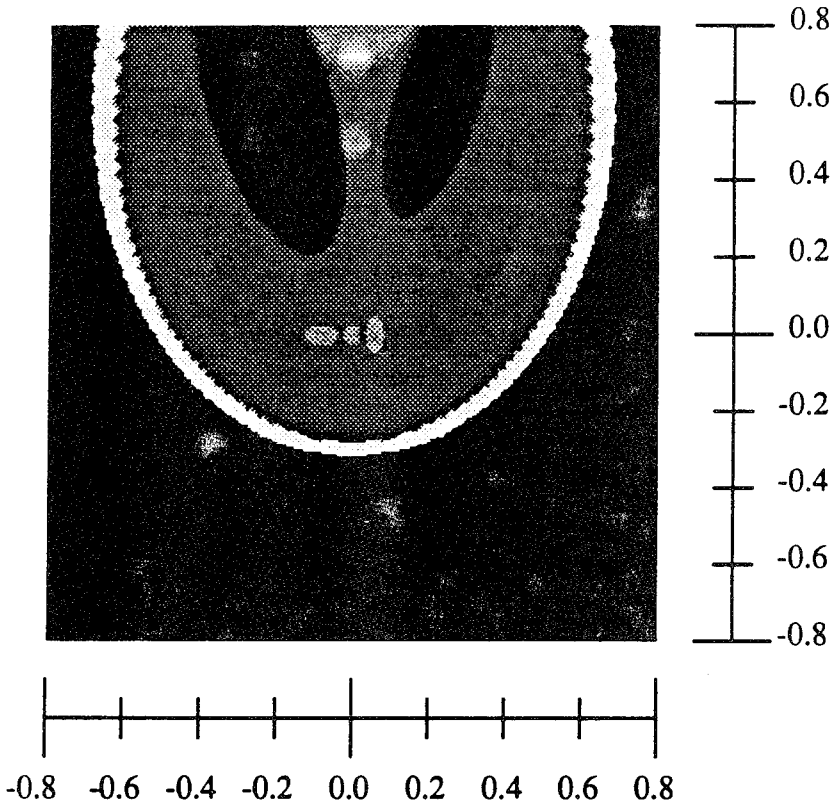


Figure 3a.

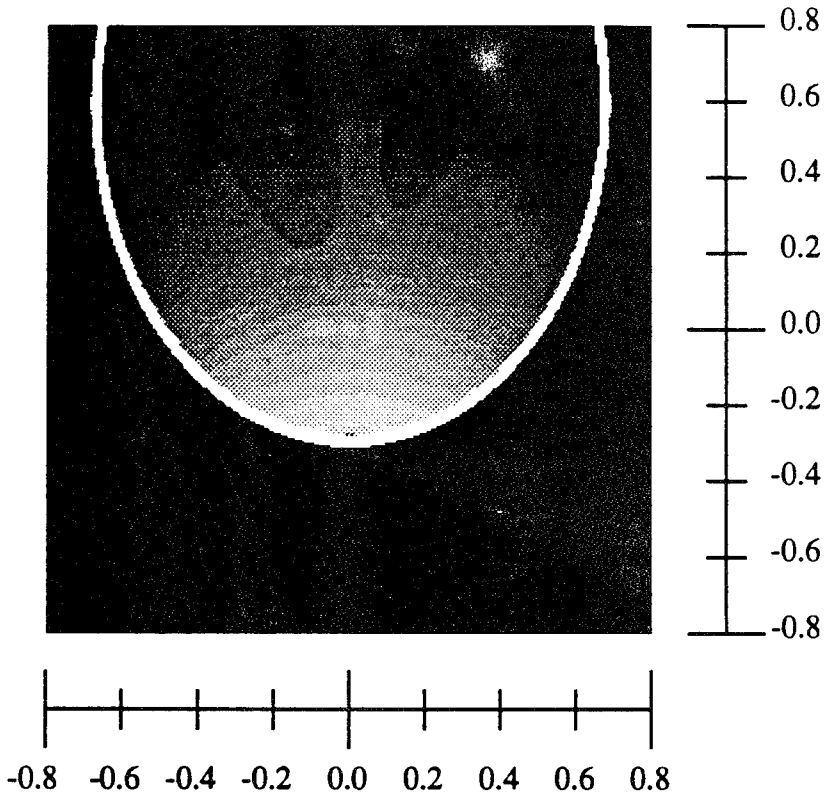


Figure 3b.

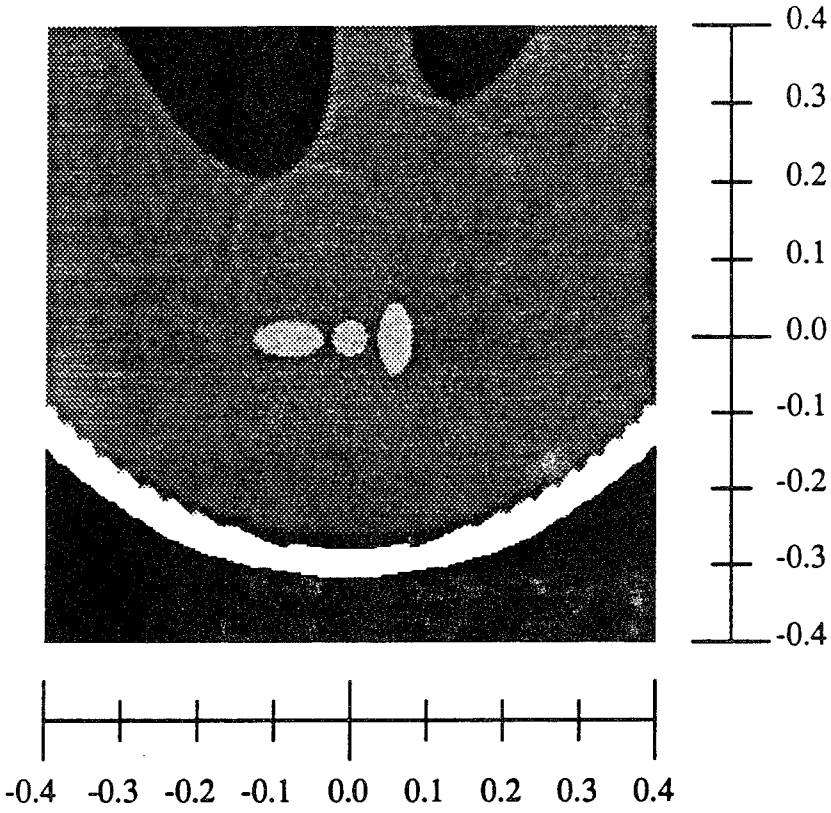


Figure 4a.

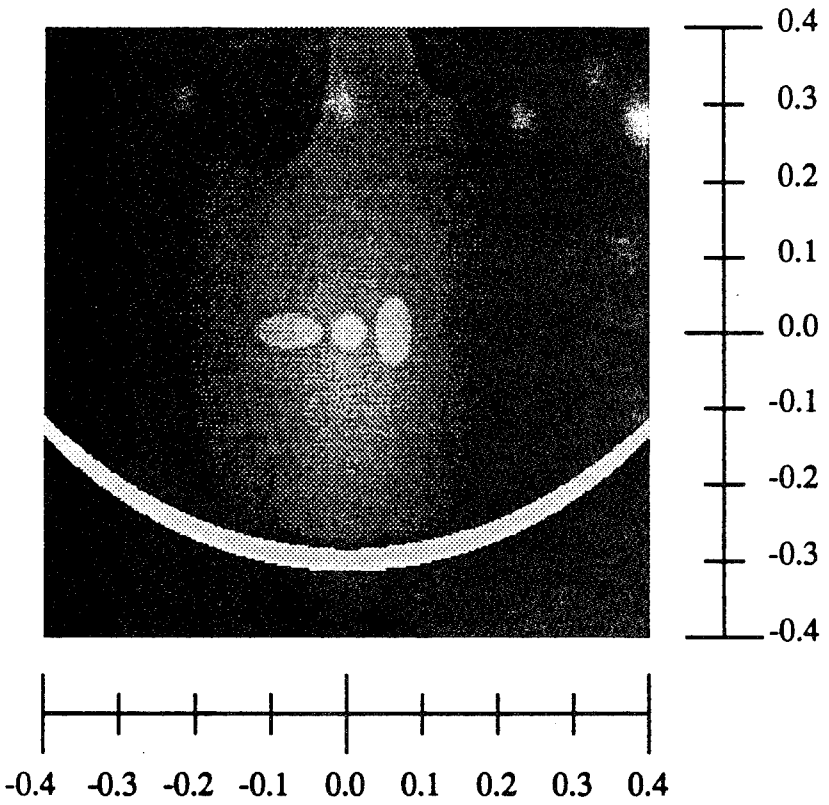


Figure 4b.

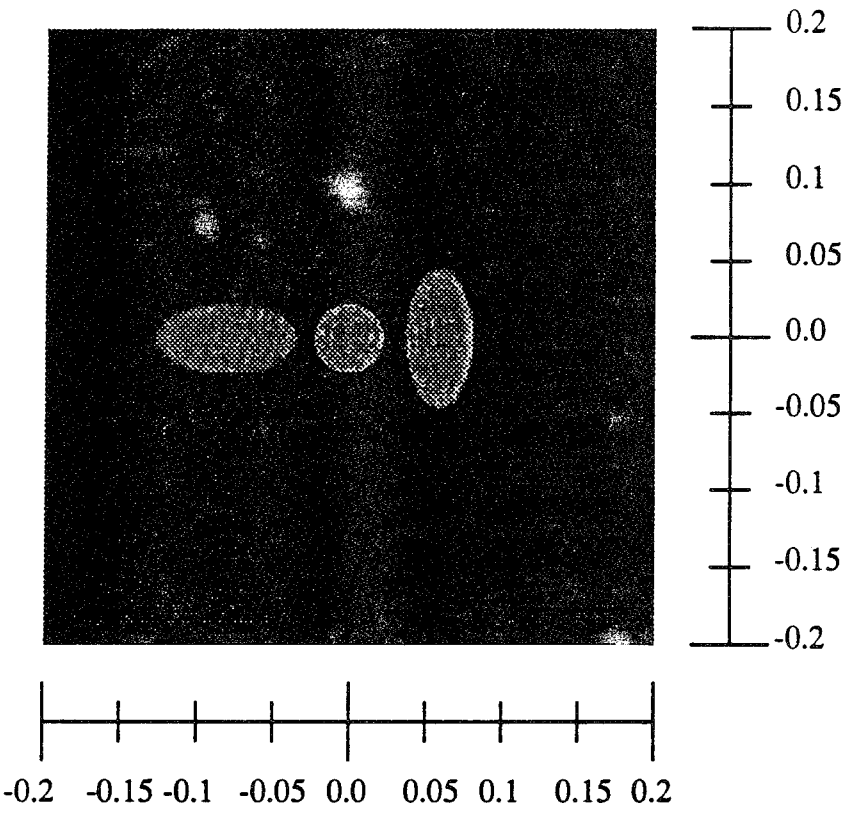


Figure 5a.

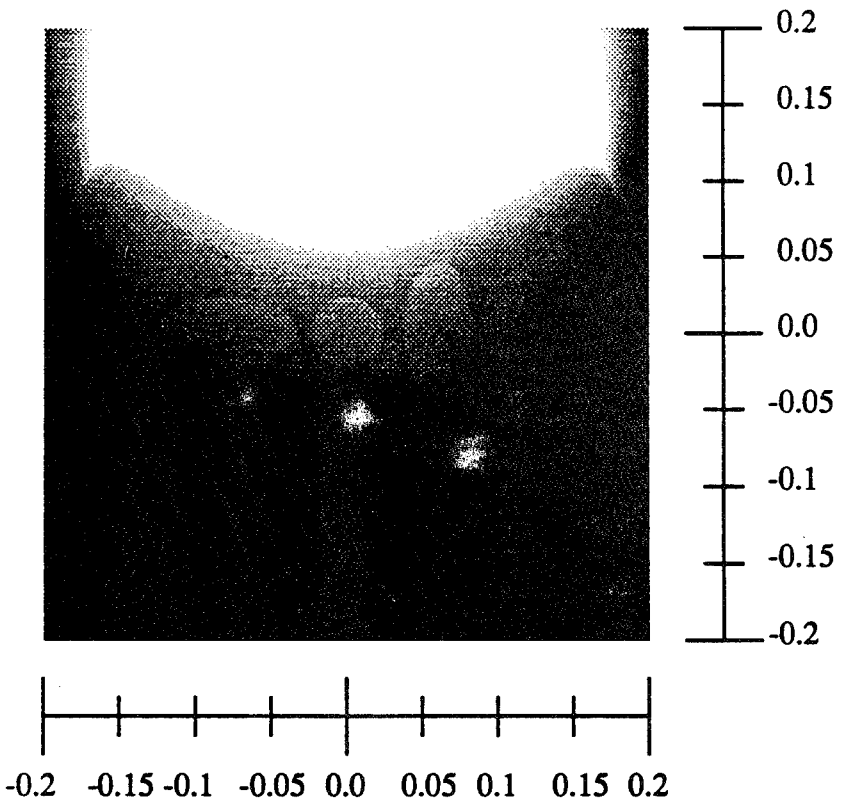


Figure 5b.

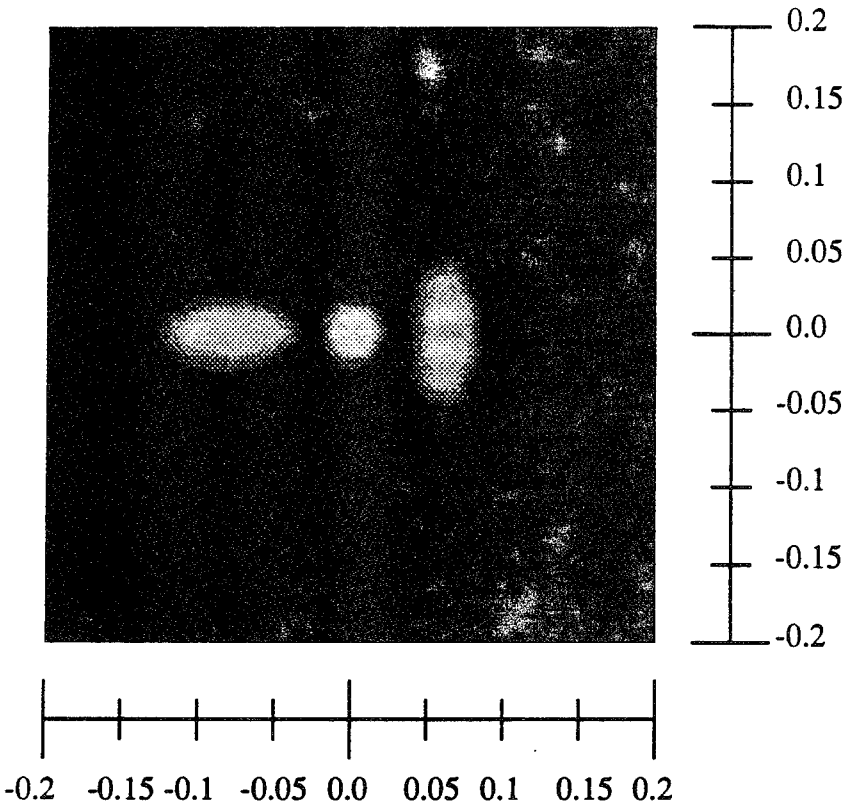


Figure 5c.

APPENDIX O

B. Sahiner and A.E. Yagle, "Reconstruction from Projections under Time-Frequency Constraints," submitted to IEEE Trans. Med. Imag., August 1993.

This paper derives a fast image-domain filter which solves the following constrained inverse Radon transform problem: Given constraints on certain wavelet coefficients of the image, compute from its projections the image which either: (a) requires the smallest perturbation of the projection data to satisfy these constraints; or (b) is the constrained linear least-squares image estimate. The wavelet transform can be used for spatially-varying filtering of an image, suppressing noise locally in smooth regions; we also discuss detection of such regions in a noisy image, which leads to the wavelet coefficient constraints. Numerical results show improvement over filtered images, since the constraints improve the reconstruction in non-constrained areas as well.

RECONSTRUCTION FROM PROJECTIONS UNDER TIME-FREQUENCY CONSTRAINTS

Berkman Sahiner and Andrew E. Yagle

Dept. of Electrical Engineering and Computer Science

The University of Michigan, Ann Arbor, Michigan 48109-2122

July 1993

Abstract

Low-pass filtering computed tomography (CT) images to reduce noise may smooth or modify image features which are very important to the physician. Image features are often more easily identified and processed in the time-frequency plane. We use time-frequency distributions for spatially-varying filtering of noisy CT images, constraining time-frequency representation coefficients of the projection data or of the reconstructed image to be zero in certain regions of the time-frequency plane. We consider two different applications: (1) filtering the projection data, and then performing image reconstruction; and (2) filtering the reconstructed image directly. Criteria minimized, subject to constraints, may be either a deterministic minimum weighted perturbation of the given projection data, or a stochastic minimum mean-square error in colored Gaussian noise. Results show improvement over processing the image with a linear spatially-invariant filter.

KEYWORDS: Mathematics of computerized tomography, image processing, inverse Radon transform algorithms, spatially-varying filtering

---

Fax:313-763-1503 email:aey@eecs.umich.edu

I. INTRODUCTION

Image reconstruction from projections in computed tomography amounts to finding the inverse Radon transform of the projection data, which is most often computed using the filtered backprojection method. A problem with the inverse Radon transform is that the ramp filter ($Q(w) \sim |w|$ in (2) below) used in the filtering stage of the filtered backprojection method amplifies the high-frequency components of the noise.

In medical images, noise usually dominates at high frequencies, i.e., the Fourier content of the noiseless image is usually small at high frequencies, whereas the Fourier content of the noise is relatively large (e.g., white noise). The high-frequency content of the image is due mostly to local "image features", such as edges. If the image does not contain features such as local sharp intensity variations or small objects embedded in noise, or if such features are unimportant, then the noise can be reduced by processing the image with a linear, spatially-invariant low-pass filter. However, in many cases, these local features are of paramount importance, and they are precisely what the physician needs; therefore, modifying them by a low-pass filter may be unacceptable. This is one reason why nonlinear and spatially-varying techniques are frequently used to process medical images [1, 2, 3, 4].

Time-frequency representations are a general framework in which perceptually significant signal features, such as the contours of an object, are both more easily identified and more easily processed [5]. Representations based solely on spatial variables do not provide readily available information about the Fourier content of the data, which is important for filtering. On the other hand, representations based solely on the Fourier analysis do not provide information about localization of significant image features. A time-frequency representation is an attempt to provide the "evolutionary spectrum" of the data, combining the advantages of representations in both domains. In this paper, we employ the discrete short-time Fourier transform (DSTFT) and the orthogonal discrete wavelet transform (DWT) as time-frequency representations. For a detailed discussion of time-frequency representations see [6], and for more discussion on the wavelet transform see [7].

For a one-dimensional function of time, a time-frequency representation is a mapping into a two-dimensional function of time and frequency, which localizes the signal energy in both time and frequency directions [8]. The two-dimensional plane on which the time-frequency representation is defined is referred to as the time-frequency plane [6]. The localization is achieved by expressing the original function in terms of basis functions which are effectively confined to a compact region of support in the time-frequency plane [9]. Conversely, for a given region in the time-frequency plane, there are a small number of functions which are effectively nonzero over that region. Compactness of basis functions helps us in identifying the signal features and filtering noise. Ideally, a signal feature is represented by as few basis functions as possible. If a feature is present at a given location in the time-frequency plane, then the basis functions which cover that location have a large contribution in the reconstruction of the signal from its time-frequency representation, i.e., these basis functions are represented by large coefficients. If the feature is absent, then the corresponding coefficients are relatively small. Thus, in a noisy situation, small coefficients are likely to have been caused by noise, rather than signal features.

A typical procedure for filtering signals in time-frequency plane makes use of the above idea: The time-frequency representation of the signal is multiplied by a one-zero mask [10, 11, 12, 13, 14]. The mask is set to unity in the region of the time-frequency plane where the energy of the signal is above a threshold (i.e., where signal features are likely to be present) and is set to zero in the region of the time-frequency plane where the signal energy is below the threshold (i.e., where signal features are unlikely to be present) [10, 13]. The filtered signal is then obtained by mapping the modified time-frequency representation back to the signal domain.

In this paper, we apply time-frequency representations to the filtering problem in computed tomography. We look at the options of filtering the projection data and filtering the reconstructed image separately. This permits incorporation of any a priori information about the image (e.g., edge locations and smooth areas) directly into the reconstruction process, rather than just post-processing the reconstructed image. We will show that this incorporation im-

proves the reconstruction of unconstrained regions as well as constrained regions. Meanwhile, the thresholding approach allows us to perform what amounts to spatially-varying low-pass filtering, in either the projection domain or the image domain.

For filtering of the projection data, we use the thresholding approach described above for two closely related time-frequency representations: the discrete short-time Fourier transform and the discrete wavelet transform. For a given projection angle, we regard the projection data as a 1-D function of the radial variable, and compute the related time-frequency representations. After time-frequency filtering is performed for each view angle separately, the filtered image is obtained using the filtered backprojection method.

For filtering the reconstructed image directly, we first use thresholding in the image domain to set constraints on the 2-D DWT of the reconstructed image. We then find the filtered image which satisfies the imposed wavelet constraints and minimizes either of two error criteria. The first criterion, which is deterministic, is the norm of the difference between the given projections and the projections of the filtered image. The second criterion, which is stochastic, is the mean-square error of the filtered image. We show that both criteria lead to similar algorithms which operate directly on the reconstructed image.

The paper is organized as follows. In Section II, we briefly review the inverse Radon transform, the DSTFT and the DWT. In Section III, we present our algorithm for filtering the projection data, and provide some examples. This extends our previous results [29, 30] from DSTFT to new results for DWT; we also present many more numerical examples. In section IV, we discuss our general approach for setting wavelet constraints in the image domain, and provide a statistical justification for thresholding the absolute value of the DWT. This extends our previous result on thresholding [27] from continuous wavelet transform to discrete wavelet transform. In Section V, we develop the deterministic formulation for processing the image, given wavelet constraints in the image domain. In Section VI, we develop the stochastic formulation, and obtain an algorithm similar to that of Section V. This links the result of Section V, which is new, to our previous results [27, 28]. In Section VII, we provide some examples using

the algorithm developed in Sections IV and V. We conclude with a summary in Section VIII.

II. INVERSE RADON TRANSFORMS, DSTFTS AND DWTS

A. *The Inverse Radon Transform*

The basic reconstruction from projections or inverse Radon transform problem is to reconstruct an image $\mu(x,y)$ from its projections $p(r,\theta)$ where

$$p(r,\theta) = \mathcal{R}\{\mu(x,y)\} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mu(x,y)\delta(r - x\cos\theta - y\sin\theta)dxdy \tag{1}$$

is the Radon transform of $\mu(x,y)$. The image is reconstructed from its projections using the inverse Radon transform

$$\mu(x,y) = \mathcal{R}^{-1}\{p(r,\theta)\} = \frac{1}{4\pi^2}\int_0^{\pi}\int_{-\infty}^{\infty} P(w,\theta)e^{jw(x\cos\theta+y\sin\theta)}Q(w)dwd\theta, \tag{2}$$

where $P(w,\theta) = \mathcal{F}_{r\to w}\{p(r,\theta)\}$ is the Fourier transform in the r variable of $p(r,\theta)$, and $Q(w) = |w|$. In practice, $Q(w)$ is a real and symmetric function of w that approximates the ideal Radon kernel $|w|$ [15, 25].

Equation (2) can be sampled in the image domain to yield a discrete image $\mu(I,J)$:

$$\mu(I,J) = \mathcal{R}_s^{-1}\{p(r,\theta)\} = \frac{1}{4\pi^2}\int_0^{\pi}\int_{-\infty}^{\infty} P(w,\theta)e^{jw(I\cos\theta+J\sin\theta)}Q(w)dwd\theta. \tag{3}$$

In practical problems, we have only samples $p(m,n)$ of $p(r,\theta)$, in which case samples $\mu(I,J)$ of the image $\mu(x,y)$ are obtained by discretizing (2) into [15]

$$\mu(I,J) = \mathcal{R}_d^{-1}\{p(m,n)\} = \frac{1}{2N}\sum_{n=0}^{N-1}\sum_{m=-\infty}^{\infty}\sum_{l=-\infty}^{\infty} q(l-m)p(m,n)h(i\cos n\Delta + j\sin n\Delta - l), \tag{4}$$

where N is the total number of views (the number of angles for which projections are available), $p(m,n)$ is the m^{th} projection in the n^{th} view, $\Delta = \pi/N$, $h(x)$ is an interpolation function, $q(m)$ is a discrete filter that approximates $q(r) = \mathcal{F}^{-1}\{Q(w)\}$ and $\mathcal{R}_d^{-1}$ defines the discrete version of the inverse Radon transform operator.

B. The DSTFT

For a discrete-time signal $f(m)$, the DSTFT is defined as [16]:

$$F(l,m) = \sum_{k=-\infty}^{\infty} f(k)a(Sm-k)e^{-j\frac{2\pi}{\mathcal{L}}kl} \tag{5}$$

where $\mathcal{L}$ is the transform size, $S \leq \mathcal{L}$ is the sampling period in the time domain, $l = 0, 1, \cdots, \mathcal{L}-1$ is the frequency variable, and m is the time variable. $a(m)$ is called the analysis filter, or the analysis window.

To compute $F(l,m)$ for a given value of the time variable m, the signal $f(k)$ is first windowed with a translated (by Sm samples) version of the analysis window $a(k)$, and then the discrete Fourier transform (DFT) is computed (this can be done exactly using an FFT). For different values of m, the analysis window is centered around different portions of the signal $f(k)$, and the DFT computes the Fourier contents of *that* portion of the windowed signal. Therefore, the time-variation in the time-frequency distribution $F(l,m)$ is specified by the amount of translation of the analysis window, and the frequency-variation is specified by the Fourier transform. Note that if $f(m)$ is nonzero over an interval of M samples, then its DSTFT will be nonzero over a lattice of approximately $\frac{M\mathcal{L}}{S}$ samples (ignoring edge effects). Since $S \leq \mathcal{L}$, the DSTFT is in general redundant.

With mild conditions on the analysis filter (e.g., the first S samples of $a(m)$ should be nonzero, or if S divides $\mathcal{L}$, none of the S polyphases of $a(m)$ should be identically zero), $f(m)$ can be recovered exactly from $F(l,m)$ [17]. The inversion formula is

$$f(m) = \sum_{k=-\infty}^{\infty} \tilde{a}(m-kS)\frac{1}{\mathcal{L}}\sum_{l=0}^{\mathcal{L}-1} F(l,m)e^{j\frac{2\pi}{\mathcal{L}}lm} \tag{6}$$

where $\tilde{a}(m)$, called the synthesis filter, is determined by the analysis filter $a(m)$. An algebraic approach to determining $\tilde{a}(m)$ for exact reconstruction is given in [18].

C. The DWT

The basic idea of the orthogonal discrete wavelet transform is to represent a sequence $f(I)$ as a superposition of translations and dilations of a wavelet $g(I)$. As opposed to the DSTFT, no

Fourier transform is computed in DWT. The frequency-variation is specified by how much the wavelet is dilated. Similarly to DSTFT, the time-variation is specified by the translation of the wavelet. The recursive formula for the wavelet decomposition $W_{2^l}f(I)$ of $f(I)$ is [21]

$$\begin{aligned} f_l(I) &= \sum_k h(2I-k)f_{l-1}(k) \\ W_{2^l}f(I) &= \sum_k g(2I-k)f_{l-1}(k) \end{aligned} \tag{7}$$

where, for a fixed scale l_0, $W_{2^{l_0}}f(I)$ is called the detail signal at scale l_0, and $f_{l_0}(I)$ is called the average signal at scale l_0. The recursion is started with $f_0(I) = f(I)$. The sequences $h(I)$ and $g(I)$ are called the scaling function and the wavelet, respectively, and satisfy

$$g(I) = (-1)^I h(1-I). \tag{8}$$

The scaling function is usually a low-pass filter, and (8) ensures that the wavelet is a high-pass filter. At each scale l, the average signal from the previous scale is convolved with the high-pass filter $g(I)$, and one sample out of two is retained. The scale number l acts like the frequency variable, and the time-index I acts as the time-variable. Note that if we start with a signal $f(I)$ which is nonzero over an interval of M samples, then the wavelet representation $\{W_{2^1}f(I), W_{2^2}f(I), \ldots, W_{2^L}f(I), f_L(I)\}$ will also have M nonzero samples (ignoring end effects due to nonzero length of $h(I)$ and $g(I)$) for any L. Thus, DWT is not redundant.

The recursive formula for the reconstruction of $f(I)$ from its wavelet transform $W_{2^l}f(I)$ is

$$f_{l-1}(I) = \sum_k h(2k-I)f_l(k) + g(2k-I)W_{2^l}f(k). \tag{9}$$

The conditions that $h(I)$ and $g(I)$ must satisfy for the orthogonal decomposition-reconstruction of (7) and (9) to work are [22]:

1. Exact reconstruction condition:

$$\sum_k h(I_1-2k)h(I_2-2k) + g(I_1-2k)g(I_2-2k) = \delta(I_1-I_2) \tag{10}$$

2. Orthogonality condition:

$$\sum_k h(k-2I_1)g(k-2I_2) = 0 \quad \text{for all } I_1 \text{ and } I_2. \tag{11}$$

For a 2-D sequence $\mu(I,J)$, the separable, orthogonal 2-D wavelet transform $W_{2^l}\mu(I,J)$ is defined recursively as

$$\begin{aligned} \mu_l(I,J) &= \sum_{k_1}\sum_{k_2} h(2I-k_1)h(2J-k_2)\mu_{l-1}(k_1,k_2) \\ W_{2^l}^{(z)}\mu(I,J) &= \sum_{k_1}\sum_{k_2} g^{(z)}(2I-k_1,2J-k_2)\mu_{l-1}(k_1,k_2), \quad z=1,2,3 \end{aligned} \tag{12}$$

where $g^{(z)}$, $z=1,2,3$ are called sub-wavelets and are defined by $g^{(1)}(I,J)= g(J)h(I)$, $g^{(2)}(I,J)= g(I)h(J)$, and $g^{(3)}(I,J)= g(I)g(J)$ [21]. Since h is low-pass and g is high-pass, the first sub-wavelet is low-pass in the I direction and high-pass in the J direction, the second is low-pass in the J direction and high-pass in the I direction, and the third is high-pass in both directions. The signals $W_{2^l}^{(z)}\mu$, $z=1,2,3$ represent details of μ in the J, I, and diagonal directions, respectively.

As opposed to the recursive formula (12), the detail signals $W_{2^l}^{(z)}\mu(I,J)$ and the average signal $\mu_l(I,J)$ can also be computed directly from $\mu(I,J)$ for any given l. The direct decomposition formula is

$$\begin{aligned} \mu_l(I,J) &= \sum_{k_1}\sum_{k_2} h_l(2^lI-k_1)h(2^lJ-k_2)\mu(k_1,k_2) \\ W_{2^l}^{(z)}\mu(I,J) &= \sum_{k_1}\sum_{k_2} g_l^{(z)}(2^lI-k_1,2^lJ-k_2)\mu(k_1,k_2). \end{aligned} \tag{13}$$

Using induction, it can easily be shown that the filters $h_l(I,J)$ $g_l^{(z)}(I,J)$ satisfy

$$\begin{aligned} h_{l+1}(I) &= \sum_{k} h_1(k)h_l(I-2^lk) \\ g_{l+1}^{(z)}(I,J) &= \sum_{k_1}\sum_{k_2} g_1^{(z)}(k_1,k_2)h_l(I-2^lk_1)h_l(J-2^lk_2) \end{aligned} \tag{14}$$

where $h_1(I)=h(I)$ and $g_1^{(z)}(I,J)=g^{(z)}(I,J)$. In words, h_{l+1} is obtained by inserting 2^l zeros between the coefficients of h_1, and convolving the resulting sequence with h_l; the average signal μ_{l+1} is then computed by convolving μ with h_{l+1} in the I and J directions and retaining one sample out of every 2^{l+1} samples in each direction. Similar remarks apply to the computation of $g_{l+1}^{(z)}$ and $W_{2^l}^{(z)}\mu$.

As a direct consequence of the orthogonality of the wavelets, it can be shown that properly translated versions of the filters $g_l^{(z)}(I,J)$ are orthogonal, i.e.,

$$\sum_{k_1}\sum_{k_2} g_l^{(z)}(2^l I - k_1, 2^l J - k_2) g_{l'}^{(z')}(2^{l'} I' - k_1, 2^{l'} J' - k_2) = \delta(l-l')\delta(z-z')\delta(I-I')\delta(J-J'). \quad (15)$$

Given the detail signals for $j = 1, \ldots, L$ and the average signal $\mu_L(I,J)$, the original signal $\mu(I,J)$ can be recovered either using a recursive formula analogous to (9) [21], or using the direct formula

$$\begin{aligned}\mu(I,J) &= \sum_{z=1}^{3}\sum_{l=1}^{L}\sum_{k_1}\sum_{k_2} g_l^{(z)}(2^l k_1 - I, 2^l k_2 - J) W_{2^l}^{(z)}\mu(k_1,k_2) \\ &+ \sum_{k_1}\sum_{k_2} h_L(2^L k_1 - I) h_L(2^L k_2 - J)\mu_L(k_1,k_2). \end{aligned} \quad (16)$$

III. PROJECTION CONSTRAINTS

A common technique for recovering a noise-corrupted signal is filtering in the Fourier plane, in which the Fourier transform of the signal is windowed to zero at frequencies where the signal energy is much smaller than the noise energy. A similar idea can be applied in the time-frequency plane: Window the time-frequency representation of the signal to zero (or constrain the time-frequency representation to be zero) in regions of the time-frequency plane where the signal energy is much smaller than the noise energy. This idea has been applied to various signals in [10, 11, 12, 13, 14], and is referred to as time-frequency filtering.

In image reconstruction from projections, filtering for noise is usually implemented in the projection domain, and is combined with ramp filtering. With a suitably chosen $q(m)$, both noise filtering and ramp filtering can be accomplished at once. The choice of $q(m)$ involves a compromise between image resolution and and noise. If the cutoff frequency of $q(m)$ is chosen to be too large, then the resulting image can be too noisy; if the cutoff frequency is chosen to be too small, then the resulting degradation of resolution may be too severe. To preserve the resolution, it is desirable to attenuate the noise energy only where the signal energy is also small. However, spatially-invariant filters are unable to change their characteristics in a

spatially-varying manner, so this selectivity cannot be attained. In contrast, time-frequency filtering is by definition spatially-varying, hence the desired selectivity may be attained by filtering the projections in the time-frequency plane.

In this section, we apply the idea of time-frequency filtering to noisy projections $p_\eta(m,n)$ of an image $\mu_a(x,y)$ by constraining the time-frequency representation of $p_\eta(m,n)$ to be zero in certain regions of the time-frequency plane. For a given view angle n_0, we regard the projection $p_\eta(m,n_0)$ as a function of m, and process its time-frequency representation. That is, m (a spatial variable) plays the role of 'time' in a time-frequency representation, and we apply time-frequency filtering to process the projection data in a spatially-varying fashion. For notational simplicity, we drop the variable n from $p_\eta(m,n)$ in the rest of this section, and refer to $p_\eta(m,n)$ as $p_\eta(m)$. The procedures described below are applied to each view separately. This section extends our previous results [29, 30] to the DWT, and provides more numerical examples.

A. Filtering using DSTFT

Let $p_a(m)$ be the noiseless projection, and $p_\eta(m)$ be the noisy projection for a given view angle n_0. To filter $p_\eta(m)$ using DSTFT, we first multiply its DSTFT $P_\eta(l,m)$ with a one-zero mask. The mask windows the DSTFT of $p_\eta(m)$ to zero in regions of the time-frequency plane ((m,l) plane) where the signal energy is much smaller than the noise energy, and does not alter the the DSTFT of $p_\eta(m)$ elsewhere. Thus we have

$$\hat{P}(l,m) = P_\eta(l,m)\mathcal{Z}(l,m) \tag{17}$$

where the one-zero mask $\mathcal{Z}(l,m)$ is either estimated from DSTFT of the data using thresholding, or is given to us as a priori information. Note that this can be viewed as a crude form of Wiener filtering, but in the time-frequency plane rather than just the frequency domain.

After $\hat{P}(l,m)$ is obtained, the second and final step is the computation of the filtered projection $\tilde{p}(m)$ whose DSTFT is $\hat{P}(l,m)$. Since DSTFT is redundant, not every 2-D sequence $\hat{P}(l,m)$ is a valid DSTFT, i.e., there may not be any sequence $\tilde{p}(m)$ whose DSTFT is $\hat{P}(l,m)$. We therefore find the signal $\tilde{p}(m)$ whose DSTFT best approximates $\hat{P}(l,m)$ in the mean-square

error sense [19]. It has been shown [20] that if the length of the analysis filter is not longer than the transform length $\mathcal{L}$, then the minimum mean-square error solution coincides with the synthesis equation (6), hence we use (6) in the examples below.

To obtain the filtered image $\tilde{\mu}(I,J)$, we perform the above procedure to compute the filtered projection for each view angle, and then proceed with the inverse Radon transform formula.

B. *Filtering using DWT*

To filter $p_\eta(m)$ using DWT, we first multiply the DWT of $p_\eta(m)$ with a mask, similarly to (17):

$$W_{2^l}\tilde{p}(m) = W_{2^l}p_\eta(m)\mathcal{Z}(l,m) \tag{18}$$

where the mask $\mathcal{Z}(l,m)$ again filters the projection in (m,l) plane. We then compute the inverse wavelet transform to find the filtered projection. In contrast to DSTFT, the DWT is not redundant; hence the filtered sequence $W_{2^l}\tilde{p}(m)$ is always a valid DWT, and there is no need to compute a minimum mean-square error solution.

C. *Examples and Discussion*

We now apply the procedures discussed above to a frequently-used phantom in medical imaging. Our phantom $\mu_a(I,J)$ (shown in Figure 1) looks like the Shepp-Logan phantom of [25]; however, the gray levels have been modified as in [26]. Reconstructions are performed from noisy projections with 128 radial samples and 128 angular samples. The additive noise is zero-mean, white, and Gaussian, with variance 6.25×10^{-4}. Using FBP and the time invariant filter $Q(w)$ from [25], with various cutoff frequencies, the "best" (in the sense of subjective human observation) reconstruction from noisy data is given in Figure 2.

For filtering using DSTFT, we determine our mask from the noiseless data as

$$\mathcal{Z}(l,m) = \begin{cases} 1 & \text{if } |P_a(l,m)|^2 \geq \nu_1 \\ 0 & \text{otherwise,} \end{cases} \tag{19}$$

where ν_1 is a fixed threshold. We choose our analysis filter to be Gaussian with length 32 (i.e., $a(m) = e^{-m^2/16}$, $m = -16, -15, \ldots, 15$; $a(m) = 0$ otherwise). The transform size $\mathcal{L} = 32$. Using DSTFT with various thresholds ν_1, the "best" filtered image $\tilde{\mu}(I, J)$ is shown in Figure 3.

For filtering using DWT, we determine the mask from the noiseless data as

$$\mathcal{Z}(l, m) = \begin{cases} 1 & \text{if } |W_{2^l} p_a(m)|^2 \geq \nu_2 \\ 0 & \text{otherwise,} \end{cases} \tag{20}$$

where ν_2 is again a fixed threshold. We use the D18 wavelet described in [22]. Using DWT with various thresholds ν_2, the "best" filtered image $\tilde{\mu}(I, J)$ is shown in Figure 4.

Comparing Figures 2, 3 and 4, we see that edges are relatively well-preserved in Figures 3 and 4 (for example, the edges of the skull are widened in Figure 2 due to low-pass filtering, but not in Figures 3 and 4). At the same time, the noise level is reduced in Figures 3 and 4, and image features are more easily distinguished from the background. The percent root mean square error, $E_{rms} = (100\%)\|\tilde{\mu} - \mu_a\|_2 / \|\mu_a\|_2$ is 15.76%, 11.44%, and 11.61% respectively for Figures 2, 3, and 4, which reflects the improvement in noise level. It should be remembered, however, that a lower E_{rms} does not necessarily mean a better image for human observation, and E_{rms} is only a secondary criterion.

Figures 5 and 6 show the analogous results when the mask is determined from the noisy data. We used a fixed threshold to obtain Figure 5, and a scale-dependent threshold to obtain Figure 6. The results are somewhat worse than Figures 3 and 4. However, note that the reconstructed image can be reprojected and time-frequency thresholding applied to reprojections. This results in an iterative algorithm that generalizes iterated Wiener filtered image reconstruction [31] from frequency domain to time-frequency plane. The results of iterated time-frequency image reconstruction will be presented elsewhere.

IV. IMAGE CONSTRAINTS

In Sections V and VI, we consider the problem of reconstructing an image from its pro-

jections, given the constraint that some fine-scale wavelet transform values around a region $\mathcal{A}$ of the image are zero. Since fine-scale wavelet transform components represent localized high-resolution features of the image, this constraint means that $\mathcal{A}$ represents a flat or slowly-varying part of the image, or that $\mathcal{A}$ is free of edges. Constraining fine-scale wavelet coefficients in $\mathcal{A}$ to be zero effectively smoothes the image, much as low-pass filtering does. The advantage of using wavelets is that this can be done on a localized basis, smoothing some areas while leaving other areas (such as edges) unaffected.

Since the Radon transform is not unitary, knowledge about one part of the image will improve the quality of the overall reconstructed image. Numerical examples will show that the constraints improve the reconstruction of the entire image, not just the constrained region.

The constraints may either be given as a priori information (for example, it may be known that $\mathcal{A}$ is free of edges), or we may impose the constraints after thresholding the absolute value of the wavelet transform of the reconstructed image. As mentioned in the introduction, the idea of thresholding the time-frequency representation of a signal for spatially-varying filtering has been previously used in the literature [13]. Below, we supply a statistical justification for thresholding the absolute value of DWT.

Detection problem formulation:

Assume that $\mu(I,J)$ is a zero-mean white Gaussian random sequence with power spectral density σ_μ^2. Let $W_{2^l}^{(z)}\mu(I,J)$, $j=1,\ldots,\infty$, $z=1,2,3$ be its wavelet transform defined using (13). Then, by the orthonormality of the wavelets (15), the quadruply indexed random sequence $W_{2^l}^{(z)}\mu(I,J)$ is uncorrelated and zero-mean, with variance σ_μ^2. To obtain a random sequence $\bar{\mu}(I,J) \neq \mu(I,J)$ whose wavelet coefficients are zero with probability one outside a region D_1, we define (compare to (16))

$$\bar{\mu}(I,J) = \sum_{l,k_1,k_2,z\in D_1} g_l^{(z)}(2^l k_1 - I, 2^l k_2 - J) W_{2^l}^{(z)}\mu(k_1,k_2) \tag{21}$$

We now state the problem. Given the noisy observations

$$\mu_\eta(I,J) = \bar{\mu}(I,J) + \eta(I,J) \tag{22}$$

of $\bar{\mu}(I,J)$, where $\eta(I,J)$ is a zero-mean white Gaussian noise sequence with power spectral density σ_η^2, determine the region D_1.

Detection problem solution

The wavelet transform of $\mu_\eta(I,J)$ is

$$W_{2^l}^{(z)}\mu_\eta(I,J) = \begin{cases} W_{2^l}^{(z)}\mu(I,J) + W_{2^l}^{(z)}\eta(I,J) & \text{if } (l,I,J,z) \in D_1 \\ W_{2^l}^{(z)}\eta(I,J) & \text{otherwise,} \end{cases} \tag{23}$$

where $W_{2^l}^{(z)}\eta(I,J)$ is the wavelet transform of $\eta(I,J)$. Note that $W_{2^l}^{(z)}\mu_\eta(I,J)$ is a zero-mean uncorrelated Gaussian random sequence whose variance is $\sigma_\eta^2+\sigma_\mu^2$ for (l,I,J,z) inside D_1 and σ_η^2 for (l,I,J,z) outside D_1. Therefore, the decision of whether a point $(l,I,J,z) \in D_1$ decouples from similar decisions for other points. Furthermore, for each point in Z^4, this becomes the well-known problem of detection of a Gaussian random variable in Gaussian noise. Its solution is the likelihood ratio test [23]

$$|W_{2^l}^{(z)}\mu_\eta(I,J)| \underset{\notin D_1}{\overset{\in D_1}{\gtrless}} \nu \tag{24}$$

where ν is the threshold (to be determined). This means that we can decide whether (l,I,J,z) is in D_1 simply by thresholding the absolute value of the wavelet transform of the noisy image at that point.

The threshold ν can be determined using the Neyman-Pearson criterion. The false alarm probability P_F=[probability of saying that $(l,I,J,z) \in D_1$ when it is not] is equated to the level of significance α, resulting in the following formula for ν:

$$\nu = \sigma_\eta \text{erfc}^{-1}(\alpha/2); \quad \text{erfc}(x) = \int_x^\infty \frac{1}{\sqrt{2\pi}} e^{-y^2/2} dy. \tag{25}$$

The detection probability P_D=[probability of correctly detecting that $(l,I,J,z) \in D_1$] is then

$$P_D = 2\text{erfc}\left(\frac{\nu}{\sqrt{\sigma_\eta^2+\sigma_\mu^2}}\right). \tag{26}$$

When the whiteness or orthonormality assumptions are relaxed, the threshold test described above is no longer guaranteed to be optimal. However, the threshold test still seems to be the "natural" approach.

In practice, we will wish to be quite sure about thresholding wavelet coefficients to zero, since these will be used as constraints in reconstructing the image from its noisy projections. Thus, $P_D \approx 1$ and the threshold ν will be very small. This has the additional advantage of requiring fewer wavelet coefficients to be constrained to zero in the algorithm of Sections V and VI.

V. IMAGE CONSTRAINTS–DETERMINISTIC APPROACH

To facilitate the derivation of our algorithm, we assume in Sections V and VI that the projections are complete, i.e., we have $p(r, \theta)$ for all r and θ. An explicit derivation for the discrete problem (sampled projections) is available in [32], but leads to results virtually identical to the continuous case.

The problem that we address in this section is defined as follows. Given projections $p(r, \theta)$ of an image $\mu_a(x, y)$, and given that the wavelet transform $W_{2^I}^{(z)}\mu_a(I, J)$ with respect to Z ($Z \leq 3$; see (12)) different sub-wavelets is zero for several values of I, J on L different scales, perturb the projections $p(r, \theta)$ such that:

1. The image reconstructed from the perturbed projections satisfies the wavelet constraints;

2. The distance between the projections and the perturbed projections is minimized.

For clarity of presentation, in Subsection V.A we consider the case where only one sub-wavelet is involved and the distance measure is the Euclidean distance. We generalize to more than one sub-wavelet, and a more general distance measure, in Subsection V.B.

A. Constraints on a Single Wavelet

Let $p(r, \theta)$ be the given projections, $\tilde{p}(r, \theta)$ be the perturbed projections, and let $\mu(I, J)$ and $\tilde{\mu}(I, J)$ be their inverse Radon transforms, respectively. Also, define the differences $\delta p(r, \theta) = p(r, \theta) - \tilde{p}(r, \theta)$ and $\delta\mu(I, J) = \mu(I, J) - \tilde{\mu}(I, J)$ and define the inner product of two real functions

$p_1(r,\theta)$ and $p_2(r,\theta)$ in the projection space as

$$< p_1(r,\theta), p_2(r,\theta) >= \int_0^{\pi} \int_{-\infty}^{\infty} p_1(r,\theta) p_2(r,\theta) dr d\theta. \tag{27}$$

Using Parseval's theorem, this inner product can be written in frequency domain as

$$< P_1(w,\theta), P_2(w,\theta) >= \frac{1}{2\pi} \int_0^{\pi} \int_{-\infty}^{\infty} P_1(w,\theta) P_2^*(w,\theta) dw d\theta \tag{28}$$

where $P_1(w,\theta)$ and $P_2(w,\theta)$ denote the continuous Fourier transforms of $p_1(r,\theta)$ and $p_2(r,\theta)$, respectively, and * denotes complex conjugate.

We assume that the wavelet transform of $\mu_a(I,J)$ with respect to the first wavelet $g^{(1)}$ is known to be zero at $C(l)$ points at scale l, where $1 \leq l \leq L$. This knowledge could come either from a priori information about the image, e.g., known absence of edges and sharp features from physiological knowledge, or from thresholding the absolute value of the wavelet transform of the image as discussed in Section IV. We therefore constrain the wavelet transform of $\tilde{\mu}(x,y)$ to be zero at those points. We index each of these points by a pair (c,l), where l denotes the scale and c enumerates the points at each scale. The constraints can then be written as

$$W_{2^l}^{(1)} \mu_a(i_{c,l}, j_{c,l}) = W_{2^l}^{(1)} \tilde{\mu}(i_{c,l}, j_{c,l}) = 0, \quad 1 \leq l \leq L, \quad 1 \leq c \leq C(l). \tag{29}$$

The problem is to determine $\tilde{p}(r,\theta)$ (or its inverse Radon transform $\tilde{\mu}(I,J)$) such that:

1. The constraints (29) are satisfied; and

2. The induced norm $\int_0^{\pi} \int_{-\infty}^{\infty} (\delta p(r,\theta))^2 dr d\theta$ is minimized.

This problem can be solved as a norm minimization problem. Taking the wavelet transform of $\delta\mu(I,J) = \mu(I,J) - \tilde{\mu}(I,J)$ using (13), the constraints can be written as

$$W_{2^l}^{(1)} \delta\mu(i_{c,l}, j_{c,l}) = \sum_I \sum_J \delta\mu(2^l i_{c,l} - I, 2^l j_{c,l} - J) g_l^{(1)}(I,J) = W_{2^l}^{(1)} \mu(i_{c,l}, j_{c,l}). \tag{30}$$

Since $\mu(I,J)$ can be computed from the given projections $p(r,\theta)$, $W_{2^l}^{(1)} \mu(i_{c,l}, j_{c,l})$ is known. Our goal is to compute $\delta\mu(I,J)$, from which the perturbed image can be computed as

$$\tilde{\mu}(I,J) = \mu(I,J) - \delta\mu(I,J). \tag{31}$$

Using the sampled inverse Radon transform formula (3), (30) can be written as

$$\frac{1}{4\pi}\int_0^{\pi}\int_{-\infty}^{\infty}\delta P(w,\theta)Q(w)G_l^{(1)}(w,\theta)\exp[jw2^l(i_{c,l}\cos\theta+j_{c,l}\sin\theta)]dwd\theta = W_{2^l}^{(1)}\mu(i_{c,l},j_{c,l}) \quad (32)$$

where $G_l^{(1)}(w,\theta)$ is defined as

$$G_l^{(1)}(w,\theta)=\sum_I\sum_J g_l^{(1)}(I,J)\exp[-jw(I\cos\theta+J\sin\theta)]. \quad (33)$$

Eq. (32) can in turn be written as an inner product using (28):

$$<\delta P(w,\theta),\frac{1}{2\pi}Q(w)G_l^{*(1)}(w,\theta)\exp[-jw2^l(i_{c,l}\cos\theta+j_{c,l}\sin\theta)]>=W_{2^l}^{(1)}\mu(i_{c,l},j_{c,l}). \quad (34)$$

Using the projection theorem, the minimum norm solution is found to be

$$\delta P(w,\theta)=\frac{1}{2\pi}\sum_{l=1}^{L}\sum_{c=1}^{C(l)}\beta_{c,l}Q(w)G_l^{*(1)}(w,\theta)\exp[-jw2^l(i_{c,l}\cos\theta+j_{c,l}\sin\theta)] \quad (35)$$

where the $\beta_{c,l}$ are computed by solving the matrix equation

$$\mathcal{M}\underline{\beta}=\underline{b}. \quad (36)$$

In (36), $\underline{\beta}$ and $\underline{b}$ are vectors which contain the unknowns $\beta_{c,l}$ and the knowns $W_{2^l}^{(1)}\mu(i_{c,l},j_{c,l})$:

$$\begin{aligned}\underline{\beta} &= [\beta_{1,1},\ldots,\beta_{C(1),1},\beta_{1,2},\ldots,\beta_{C(2),2},\ldots,\beta_{C(L),L}]^T\\ \underline{b} &= [W_{2^1}^{(1)}\mu(i_{1,1},j_{1,1}),\ldots,W_{2^1}^{(1)}\mu(i_{C(1),1},j_{C(1),1}),W_{2^2}^{(1)}\mu(i_{1,2},j_{1,2}),\ldots,\\ &\quad W_{2^2}^{(1)}\mu(i_{C(2),2},j_{C(2),2}),\ldots W_{2^L}^{(1)}\mu(i_{C(L),L},j_{C(L),L})]^T.\end{aligned} \quad (37)$$

The system matrix $\mathcal{M}$ consists of L^2 submatrices $\mathcal{M}_{l,m}$, each of which contains the inner product of the terms which are at the right-hand side of the inner product in (34). That is,

$$\mathcal{M}=\begin{bmatrix}\mathcal{M}_{1,1} & \mathcal{M}_{1,2} & \cdots & \mathcal{M}_{1,L}\\ \mathcal{M}_{2,1} & \mathcal{M}_{2,2} & \cdots & \mathcal{M}_{2,L}\\ \vdots & \vdots & & \vdots\\ \mathcal{M}_{L,1} & \mathcal{M}_{L,2} & \cdots & \mathcal{M}_{L,L}\end{bmatrix}, \quad (38)$$

where the $(u,v)^{th}$ entry of $\mathcal{M}_{l,m}$ is

$$< \frac{1}{2\pi} Q(w) G_l^{*(1)}(w,\theta) \exp[-jw2^l(i_{u,l}\cos\theta + j_{u,l}\sin\theta)],$$
$$\frac{1}{2\pi} Q(w) G_m^{*(1)}(w,\theta) \exp[-jw2^m(i_{v,m}\cos\theta + j_{v,m}\sin\theta)] > . \quad (39)$$

Writing the inner product (39) as an integral, expanding $G_m^{*(1)}(w,\theta)$ and $G_l^{*(1)}(w,\theta)$ using (33), and defining $R(I,J) = \mathcal{R}_s^{-1}\{q(r)\}$, we find that the $(u,v)^{th}$ entry of $\mathcal{M}_{l,m}$ is given by

$$\frac{1}{2\pi} \sum_{s_1} \sum_{t_1} \sum_{s_2} \sum_{t_2} g_l^{(1)}(s_1,t_1) g_m^{(1)}(s_2,t_2) R(2^m i_{v,m} - 2^l i_{u,l} + s_1 - s_2, 2^m j_{v,m} - 2^l j_{u,l} + t_1 - t_2). \quad (40)$$

The inverse Fourier transform of $\delta P(w,\theta)$ is computed by finding the sampled inverse Radon transform of (35), which is

$$\delta\mu(I,J) = \sum_{l=1}^{L} \sum_{c=1}^{C(l)} \beta_{c,l} y_l^{(1)}(2^l i_{c,l} - I, 2^l j_{c,l} - J), \quad (41)$$

where $y_l^{(1)}$ is defined in terms of the convolution of $g_l^{(1)}$ and R as

$$y_l^{(1)}(I,J) = \frac{1}{2\pi} \sum_{s_1} \sum_{t_1} g_l^{(1)}(s_1,t_1) R(I - s_1, J - t_1). \quad (42)$$

Thus, to compute the perturbed (filtered) image, we first compute $\delta\mu(I,J)$ using (41), where $\beta_{c,l}$ are solved from the linear system of equations (36), and $y_l^{(1)}$ is given by (42). Then, the perturbed image is found using (31).

Note that if the wavelet $g(I)$ is a finite-length sequence, a constraint on the wavelet transform of the image will involve only a finite number of pixel values. However, the perturbed image will be improved for *all* I and J, since the inverse Radon transform $\mathcal{R}_s^{-1}\{q(r)\}$ (and thus $\delta\mu(I,J)$ computed using (41)) will have infinite extent for a general choice of the filter $Q(w)$. This is illustrated in the numerical examples of Section VII.

B. Constraints on Several Sub-Wavelets

We now generalize to more than one sub-wavelet, and to a more general distance measure. Let $Z \leq 3$ be the maximum number of wavelets used, and let $z \in 1,\ldots,Z$ denote the sub-wavelet

number (see (12)). We index points by the superscript z and subscripts (c,l), where l denotes the scale and c enumerates the points at each scale. The constraints are

$$W_{2^l}^{(z)}\tilde{\mu}(i_{c,l}^{(z)}, j_{c,l}^{(z)}) = 0, \quad 1 \le l \le L, \quad 1 \le c \le C(l), \quad 1 \le z \le Z. \tag{43}$$

We also now redefine the inner product of two functions $P_1(w,\theta)$ $P_2(w,\theta)$ as

$$< P_1(w,\theta), P_2(w,\theta) >= \frac{1}{2\pi}\int_0^{\pi}\int_{-\infty}^{\infty}\frac{P_1(w,\theta)P_2^{*}(w,\theta)}{T(w)}dwd\theta \tag{44}$$

where $T(w)$ is the Fourier transform of a weighting function $t(r)$. $T(w)$ can be any real, even and positive function of w.

The inner product (44) induces a generalized, *weighted* distance, (compare to $\int_0^{\pi}\int_{-\infty}^{\infty}(\delta p(r,\theta))^2 drd\theta$) which allows the projections $p(r,\theta)$ to be weighted non-uniformly in r. This is useful in situations where some projections are known more reliably than others; such situations can arise in many different ways in collecting medical imaging data. For example, if the projections are known to be noisier for high $|w|$, due to the limited radial resolution in a rotating CAT-scan, these values can be assigned less weight in minimizing the perturbation of the projections (i.e., they can be perturbed more without much increase in penalty).

The arguments used in the previous subsection lead to the formula

$$\delta\mu(I,J) = \sum_{z=1}^{Z}\sum_{l=1}^{L}\sum_{c=1}^{C(l)}\beta_{c,l}^{(z)}yt_l^{(z)}(2^l i_{c,l}^{(z)} - I, 2^l j_{c,l}^{(z)} - J), \tag{45}$$

where

$$yt_l^{(z)}(I,J) = \frac{1}{2\pi}\sum_{s_1}\sum_{t_1}g_l^{(z)}(s_1,t_1)Rt(I-s_1, J-t_1), \tag{46}$$

and

$$Rt(I,J) = \mathcal{R}_s^{-1}\{q * t\} = \mathcal{R}_s^{-1}\left\{\int_{-\infty}^{\infty}q(\varsigma)t(r-\varsigma)d\varsigma\right\}. \tag{47}$$

The $\beta_{c,l}^{(z)}$ are again computed by solving $\mathcal{M}\underline{\beta} = \underline{b}$, where

$$\mathcal{M} = \begin{bmatrix} \mathcal{M}_{1,1} & \mathcal{M}_{1,2} & \cdots & \mathcal{M}_{1,L} \\ \mathcal{M}_{2,1} & \mathcal{M}_{2,2} & \cdots & \mathcal{M}_{2,L} \\ \vdots & \vdots & & \vdots \\ \mathcal{M}_{L,1} & \mathcal{M}_{L,2} & \cdots & \mathcal{M}_{L,L} \end{bmatrix} \qquad \mathcal{M}_{l,m} = \begin{bmatrix} \mathcal{M}_{l,m}(1,1) & \cdots & \mathcal{M}_{l,m}(1,Z) \\ \vdots & & \vdots \\ \mathcal{M}_{l,m}(Z,1) & \cdots & \mathcal{M}_{l,m}(Z,Z) \end{bmatrix} \tag{48}$$

and where the $(u,v)^{th}$ entry of $\mathcal{M}_{l,m}(z_1,z_2)$ is

$$\frac{1}{2\pi}\sum_{s_1}\sum_{t_1}\sum_{s_2}\sum_{t_2} g_l^{(z_1)}(s_1,t_1)g_m^{(z_2)}(s_2,t_2)Rt(2^m i_{v,m}^{(z_1)}-2^l i_{u,l}^{(z_2)}+s_1-s_2, 2^m j_{v,m}^{(z_1)}-2^l j_{u,l}^{(z_2)}+t_1-t_2). \quad (49)$$

VI. IMAGE CONSTRAINTS–STOCHASTIC APPROACH

In this section, we address the following *stochastic* version of the problem addressed in Section V: Suppose that we are given noisy observations $p_\eta(r,\theta) = p_a(r,\theta) + \eta(r,\theta)$ of the projections $p_a(r,\theta)$ of an actual image $\mu_a(x,y)$, where $\eta(r,\theta)$ is a zero-mean Gaussian random process in r for each θ, and is uncorrelated in θ. Given constraints on the wavelet transform values of the image, find the image $\tilde{\mu}(I,J)$ such that:

1. $\tilde{\mu}(I,J)$ satisfies the constrained image values; and

2. $E\{\sum_I \sum_J (\mu_a(I,J) - \tilde{\mu}(I,J))^2\}$ is minimized.

As in the previous section, we first consider constraints on a single sub-wavelet. Also, to draw a parallel with Subsection V.A, we assume in Subsection VI.A that the additive noise in the projections is white in each slice in r, i.e., $E[\eta(r_1,\theta_1)\eta(r_2,\theta_2)] = \delta(r_1-r_2)\delta(\theta_1-\theta_2)$. We generalize to non-white noise, and several sub-wavelets, in Subsection VI.B.

A. Constraints on a Single Wavelet

Let $\mu_\eta(I,J) = \mathcal{R}_s^{-1}\{p_\eta(r,\theta)\}$, and $\epsilon(I,J) = \mathcal{R}_s^{-1}\{\eta(r,\theta)\}$ so that

$$\mu_\eta(I,J) = \mu_a(I,J) + \epsilon(I,J). \quad (50)$$

To solve the problem, we first compute an estimate $\hat{\epsilon}(I,J)$ of $\epsilon(I,J)$ using the given constraints, and then we subtract the noise estimates from the noisy image $\mu_\eta(I,J)$.

Let the constraints be as given in (29). Then, by taking the wavelet transform of both sides of (50), we find (compare to (30))

$$W_{2^l}^{(1)}\epsilon(i_{c,l}, j_{c,l}) = \sum_I \sum_J \epsilon(2^l i_{c,l} - I, 2^l j_{c,l} - J) g_l^{(1)}(I,J) = W_{2^l}^{(1)}\mu_\eta(i_{c,l}, j_{c,l}). \quad (51)$$

As in (30), the $\mu_\eta(I,J)$ can be computed from the given projections $p_\eta(r,\theta)$, so the $W_{2^l}^{(1)}\mu_\eta(i_{c,l},j_{c,l})$ are known. Our goal is to compute $\hat{\epsilon}(I,J)$, from which the image $\tilde{\mu}(I,J)$ can be computed as (compare to (31))

$$\tilde{\mu}(I,J) = \mu_\eta(I,J) - \hat{\epsilon}(I,J). \tag{52}$$

Since $\eta(r,\theta)$ is zero-mean and jointly Gaussian in r and θ, $\epsilon(I,J)$ (which is a linear combination of $\eta(r,\theta)$) is also zero-mean and jointly Gaussian in I and J. The solution to the problem of finding $\hat{\epsilon}(I,J)$ is the linear minimum mean-square (LMMSE) estimate

$$\hat{\epsilon}(I,J) = \sum_{l=1}^{L}\sum_{c=1}^{C(l)} \beta_{c,l} E[\epsilon(I,J) W_{2^l}^{(1)} \epsilon(i_{c,l},j_{c,l})], \tag{53}$$

where the $\beta_{c,l}$ are computed by solving the matrix equation (compare to (36))

$$\mathcal{M}\underline{\beta} = \underline{b}. \tag{54}$$

As in (36), $\underline{\beta}$ and $\underline{b}$ are vectors which contain the unknowns $\beta_{c,l}$ and the knowns $W_{2^l}^{(1)}\mu_\eta(i_{c,l},j_{c,l})$. The system matrix $\mathcal{M}$ is given by (38), where the $(u,v)^{th}$ entry of $\mathcal{M}_{l,m}$ is

$$\mathcal{M}_{l,m}(u,v) = E[W_{2^l}^{(1)}\epsilon(i_{u,l},j_{u,l}) W_{2^m}^{(1)}\epsilon(i_{v,m},j_{v,m})]. \tag{55}$$

If the noise in the projections is white in r, i.e., if $E[\eta(r_1,\theta_1)\eta(r_2,\theta_2)] = \delta(r_1-r_2)\delta(\theta_1-\theta_2)$, then it can easily be shown [24] that the autocorrelation of $\epsilon(I,J)$ is given by

$$R_\epsilon(I,J) = E[\epsilon(I',J')\epsilon(I+I',J+J')] = \frac{1}{2\pi}\mathcal{R}_s^{-1}\{q(r)\} = \frac{1}{2\pi}R(I,J). \tag{56}$$

As a consequence, we find that

$$E[\epsilon(I,J) W_{2^l}^{(1)}\epsilon(i_{c,l},j_{c,l})] = y_l^{(1)}(2^l i_{c,l} - I, 2^l j_{c,l} - J) \tag{57}$$

and

$$E[W_{2^l}^{(1)}\epsilon(i_{u,l},j_{u,l}) W_{2^m}^{(1)}\epsilon(i_{v,l},j_{v,l})] = \frac{1}{2\pi}\sum_{s_1}\sum_{t_1}\sum_{s_2}\sum_{t_2}$$
$$g_l^{(1)}(s_1,t_1) g_m^{(1)}(s_2,t_2) R(2^m i_{v,m} - 2^l i_{u,l} + s_1 - s_2, 2^m j_{v,m} - 2^l j_{u,l} + t_1 - t_2), \tag{58}$$

which is the same as (40).

We thus find that $\hat{\epsilon}(I,J)$ computed using (53) coincides with $\delta\mu(I,J)$ computed using (41).

B. *Constraints on Several Sub-Wavelets*

We now assume that the constraints are given by (43), and that the autocorrelation of the additive noise in the projections is no longer white, but is given by

$$E[\eta(r_1,\theta_1)\eta(r_2,\theta_2)] = t(r_1 - r_2)\delta(\theta_1 - \theta_2). \tag{59}$$

That is, the noise is still uncorrelated between projection angles, but correlated in r at a single angle. Then, using the arguments of the previous subsection, it is easily shown that the noise estimate of the image $\hat{\epsilon}(I,J)$ is the same as $\delta\mu(I,J)$ computed using (41), i.e.,

$$\hat{\epsilon}(I,J) = \sum_{z=1}^{Z}\sum_{l=1}^{L}\sum_{c=1}^{C(l)} \beta_{c,l}^{(z)} yt_l^{(z)}(2^l i_{c,l}^{(z)} - I, 2^l j_{c,l}^{(z)} - J), \tag{60}$$

where $yt_l^{(z)}(I,J)$ is given by (46) and $\beta_{c,l}^{(z)}$ are computed by solving $\mathcal{M}\underline{\beta} = \underline{b}$, where $\mathcal{M}$ is given by (48) and (49).

C. *Relation Between Image Constraints and Projection Constraints*

The discretized inverse Radon transform (4) can be regarded as an operation of multiplying a vector of projection data by a matrix, yielding a vector of image pixels. Similarly, the DSTFT (5) and the DWT (12), along with their inverse transforms (6) and (16), can also be regarded as operations of multiplying a vector of transform values by a matrix, yielding a vector of image pixels (or vice-versa).

As a result, it is clear that constraining certain values of the DWT of the image is equivalent to constraining linear combinations of the DWT of the projection data, and vice-versa. That is, the real difference between Section III (in which DWTs of projection data were constrained) and Section V (in which DWTs of the image were constrained) is in the specific constraints applied, and their physical interpretations. The basic underlying concepts, with least-squares criteria, are identical. This unifies the two problems.

VII. IMAGE CONSTRAINTS–EXAMPLES AND DISCUSSION

In this section, we present three numerical examples which illustrate the results of Sections V and VI. Since the stochastic and deterministic developments lead to the same formula, our examples will be formulated in terms of the stochastic (noisy) case only. The examples are the same as those which appeared in [27].

As mentioned in the beginning of Section V, the algorithms developed in the previous two sections were based on complete projections. In practice, only samples $p(m,n)$ of projections are available. This does not present any difficulties, and the only modification to our algorithm is the replacement of (47) by

$$Rt(I,J) = \mathcal{R}_d^{-1}\{q * t\} = \mathcal{R}_d^{-1}\left\{\sum_{t_1} q(t_1)t(m-t_1)\right\}, \tag{61}$$

where $q(m)$ is a discrete filter that approximates $q(r)$, and $t(m)$ is the radial autocorrelation function of the discrete noise in the projections.

Example 1: The noiseless image used in this example is a disk of value 1.00 and radius 0.81 in a background of value 0. Projections of the disk are computed over 128 angles and 128 lines in each angle. The reconstruction from noiseless projections is shown in Figure 7. The noise added to the projections is obtained by passing zero-mean white Gaussian noise with variance 0.01 through a filter whose discrete-time Fourier transform is $(sin(w))^{32}$. The 100×100 image obtained from the noisy projections using FBP is shown in Figure 8.

The wavelets we use are two sub-wavelets of the Haar basis, which can be regarded as difference operators in the I and J directions (the third Haar sub-wavelet, which can be regarded as a difference operator in the diagonal direction, is not used). We constrain the two finest-scale wavelet coefficients to be zero in a 15×55 rectangular area $\mathcal{A}_0$ inside the disk. We use (60) to estimate the noise $\hat{\epsilon}(i,j)$, where the matrix $\mathcal{M}$ is given by (48).

The MMSE image is shown in Figure 9. Table 1 shows the average performance of our procedure for 10 different noise realizations for this example. The area obtained by enlarging $\mathcal{A}_0$ by 20 pixels in every direction is denoted as $\mathcal{A}_1$; this is roughly the area in which we expect

improvement, due to the support of the filter $yt(I, J)$. The whole image is denoted by $\mathcal{A}_t$. From Table 1, we see that the noise in $\mathcal{A}_0$ is almost completely eliminated. We also find that, compared to the unprocessed noisy image, the noise powers in the regions $\mathcal{A}_1 - \mathcal{A}_0$ and $\mathcal{A}_t - \mathcal{A}_0$, *in which we do not have any wavelet constraints*, are reduced by 13.1% and 6.4%, respectively. This shows that constraining wavelet coefficients in a given region *improves the reconstruction in other regions*. This is because the noise ϵ in the reconstructed image is non-white, due to the fact that the Radon transform is non-unitary and the additive noise η on the projections is non-white.

Example 2: In the second example, the original image is not flat in $\mathcal{A}_0$, but varies smoothly in that region. The noiseless image, which is the union of a disk and an exponential, is shown in Figure 10. The additive noise is the same as in Example 1, and the noisy reconstruction is shown in Figure 11. The wavelet constraints used are the same as in Example 1. Since the noiseless image has small, but not negligible, high-resolution components, we need to use a wavelet which has wider support than the Haar wavelet in this example. Otherwise, setting the fine-scale wavelet coefficients to zero may cause "blocking artifacts" in the image. The result of our algorithm with the six-coefficient Daubechies wavelet [22] is shown in Figure 12. Note again that noise has been reduced not only in the constrained region, but in other regions.

Example 3: In this example, we use the original Shepp-Logan head phantom [25] as our noiseless image. The autocorrelation of the noise is the same as previous examples, and the noise variance is 4×10^{-6}. The 128×128 noisy image is shown in Figure 13. We use the Haar wavelet, and constrain the two finest-scale wavelet coefficients to be zero over a region D of the image. The region D is obtained by the thresholding approach over a region D' in the center of the image. First, the second-finest wavelet coefficients inside D' are set to zero whenever their absolute value is below a threshold. The region which will be affected by the above operation is called D''. Then, inside D'', another threshold is used to set the finest-scale coefficients to zero. The resulting MMSE image is shown in Figure 14. The noise power has been reduced by 20.3% in the whole image, while still preserving the edges.

VIII. CONCLUSION

We have shown how reconstruction of CT images can be improved by time-frequency filtering. We have considered two different filtering strategies. The first strategy is filtering the projection data and then computing the filtered image using filtered backprojection. The second strategy is filtering the reconstructed image directly. In the first strategy, the DSTFT or the DWT of the projections are set to zero in certain areas of the time-frequency plane, in effect performing a spatially-varying low-pass filtering of the data. In the second strategy, the DWT of the reconstructed image is constrained to be zero in certain areas of the time-frequency plane, and the reconstructed images satisfying these constraints and minimizing either the deterministic least-squares perturbation of the projections, or the stochastic mean-square error, is computed. These criteria were then expanded to a weighted deterministic perturbation, or a stochastic minimum mean-square estimate in colored Gaussian noise. Examples using both schemes show that time-frequency filtering can produce results superior to linear, spatially-invariant filtering for CT images.

REFERENCES

[1] L. Keselbrener, Y. Shimoni, S. Akselrod, "Nonlinear filters applied on computerized axial tomography–Theory and phantom images," Medical Physics, vol. 19, pp. 1057-1064, 1992.

[2] M. Okada, "Noise evaluation and filter design in CT images," IEEE Trans. Biom. Eng., vol. 32, pp. 713-719, 1985.

[3] W. E. Higgins, "A flexible implementation of maximum homogeneity filtering for 2-D and 3-D images," IEEE Trans. Sig. Proc., vol. 39, pp. 2325-2331, 1991.

[4] J. B. Weaver, Y. S. Xu, D. M. Healy, L. D. Cromwell, "Filtering noise from images with wavelet transforms," Magnetic Resonance in Medicine, vol. 21, pp. 288-295, 1991.

[5] S. Mallat and W. L. Hwang, "Singularity detection and processing with wavelets," IEEE Trans. Inform. Theory, vol. 38, pp. 617-643, 1992.

[6] L. Cohen, "Time-frequency distributions - a review," Proc. IEEE, vol. 77, pp. 941-981, 1989.

[7] O. Rioul and M. Vetterli, "Wavelets and signal processing," IEEE Signal Processing Magazine, pp. 14-38, Oct. 1991.

[8] J. Jeong, W. W. Williams, "Kernel Design for reduced interference distributions," IEEE Trans. Sig. Proc., vol. 40, pp. 402-412, 1992.

[9] I. Daubechies, "The wavelet transform, time-frequency localization and signal analysis," IEEE Trans. Inform. Theory, vol. 36, pp. 961-1005, 1990.

[10] T. E. Koczwara and D.L. Jones, "On mask selection for time-varying filtering using the Wigner Distribution," in Proc. ICASSP-90, pp. 2487-2490.

[11] J. Jeong and W. J. Williams, "Time-varying filtering and signal synthesis," in *Time-Frequency Signal Analysis*, B. Boashash, Ed., Melbourne: Longman and Cheshire, 1991.

[12] B. Boashash and L. B. White, "Instantaneous frequency estimation and automatic time-varying filtering," in Proc. ICASSP-1990, pp. 1221-1224.

[13] M. Bikdash and K. B. Yu, "Linear shift varying filtering of non-stationary chirp signals," in Proc. IEEE Southeastern Symposium on System Theory, 1988, pp. 428-432.

[14] G. F. Boudreaux-Bartels and T. W. Parks, "Time-varying filtering and signal estimation using Wigner distribution synthesis techniques," IEEE Trans. Acoust., Speech, Sig. Proc., vol. 34, pp.442-451, 1988.

[15] S. W. Rowland, "Computer implementation of image reconstruction formulas", in *Image Reconstruction from Projections, Implementation and Applications.* Ed. G. T. Herman, New York: Springer, 1978.

[16] M. R. Portnoff, "Representation of digital signals and systems based on the short-time Fourier transform," IEEE Trans. Acoust., Speech, Sig. Proc., vol. 28, pp. 55-69, 1980.

[17] Z. Shpiro and D. Malah, "An algebraic approach to discrete short-time Fourier transform analysis and synthesis," in Proc. ICASSP-84, pp. 2.3.1-2.3.4, pp. 804-807.

[18] Z. Shpiro and D. Malah, "Design of filters for discrete short-time Fourier transform synthesis," in Proc. ICASSP-85, pp. 14.6.1-14.6.4, pp. 537-540.

[19] B. E. A. Saleh and N. S. Subotic, "Time-variant filtering of signals in the mixed time-frequency domain," IEEE Trans. Acoust., Speech, Sig. Proc., vol. 33, pp.1479-1487, 1985.

[20] A. Dembo and D. Malah, "Signal synthesis from modified discrete short-time Fourier transform," IEEE Trans. Acoust., Speech, Sig. Proc., vol. 34, pp.168-180, 1988.

[21] S. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," IEEE Trans. Patt. Anal. Machine Intell., vol. PAMI-11, pp. 674-693, 1989.

[22] I. Daubechies, "Orthonormal bases of compactly supported wavelets," Comm. in Pure and Applied Math., vol. 41, pp. 909-996, 1988.

[23] H. L. Van Trees, *Detection, Estimation, and Modulation Theory.* New York: Wiley, 1968.

[24] A. K. Jain and S. Ansari, "Radon transform theory for random fields and optimum image reconstruction from noisy projections," in Proc. ICASSP 1984, pp. 12A.7.1-12A.7.4.

[25] L. A. Shepp and B. F. Logan, "The Fourier reconstruction of a head section," IEEE Trans. Nucl. Sci., vol. NS-21, pp. 21-42, 1974.

[26] S. X. Pan and A. C. Kak, "A computational study of reconstruction algorithms for diffraction tomography: Interpolation versus filtered backpropagation," IEEE Trans. Acoust., Speech, Sig. Proc., vol. ASSP-31, pp. 1262-1275, 1983.

[27] B. Sahiner and A. E. Yagle, "Image reconstruction from projections under wavelet constraints," IEEE Trans. Sig. Proc., Special Issue on Wavelets and Signal Processing, to appear Dec. 1993.

[28] B. Sahiner and A. E. Yagle, "On the use of wavelets in inverting the Radon transform," in Proc. IEEE Medical Imaging Conf., 1992, pp. 1129-1131.

[29] B. Sahiner and A. E. Yagle, "Time-frequency distribution inversion of the Radon transform," to be published in IEEE Trans. Image Proc., Oct. 1993.

[30] B. Sahiner and A. E. Yagle, "Time-frequency distribution inversion of the Radon transform," in Proc. IEEE Medical Imaging Conf., 1991, pp. 2043-2047.

[31] W. Y. Sun, S. E. Derenzo, and T. F. Budinger, "An adaptive Wiener filter for PET based on a novel technique for spectral estimation," in Proc. IEEE Medical Imaging Conf., 1991, pp. 1933-1937.

[32] B. Sahiner and A. E. Yagle, "Constrained image reconstruction from projections using the wavelet transform," Tech. Report, Dept. of EECS, The Univ. of Mich., Ann Arbor, MI, Aug. 1992.

Average of 10 noise realizations with σ^2=0.01

	Noise Power in			% Improvement in	
	R_0	R_1	R_t	$R_1 - R_0$	$R_t - R_0$
Noisy Image	5.20	28.48	53.35		
Example 1	0.082	20.30	45.15	13.15	6.40

Table 1

Figure Headings

Fig. 1. Noiseless Shepp-Logan phantom

Fig. 2. Best reconstruction from noisy data and linear time-invariant filter.

Fig. 3. The result of filtering with DSTFT, the mask is obtained from noiseless projections.

Fig. 4. The result of filtering with DWT, the mask is obtained from noiseless projections.

Fig. 5. The result of filtering with DSTFT, the mask is obtained from noisy projections.

Fig. 6. The result of filtering with DWT, the mask is obtained from noisy projections.

Fig. 7. The reconstruction of the disk used in Example 1 from noiseless projections.

Fig. 8. The noisy image for Example 1, obtained from the noisy projections by FBP.

Fig. 9. The MMSE image obtained by constraining the two finest-scale wavelet coefficients in $\mathcal{A}_0$ to 0.

Fig. 10. The reconstruction of the image used in Example 2 from noiseless projections.

Fig. 11. The noisy image for Example 2, obtained from the noisy projections by FBP.

Fig. 12. The MMSE image obtained by constraining the two finest-scale wavelet coefficients in $\mathcal{A}_0$ to 0; the wavelet basis function is the 6 coefficient Daubechies wavelet; Example 2.

Fig. 13. The noisy image for Example 3, obtained from the noisy projections of the Shepp-Logan phantom.

Fig. 14. MMSE image obtained by constraining the wavelet coefficients of the noisy image; Example 3.

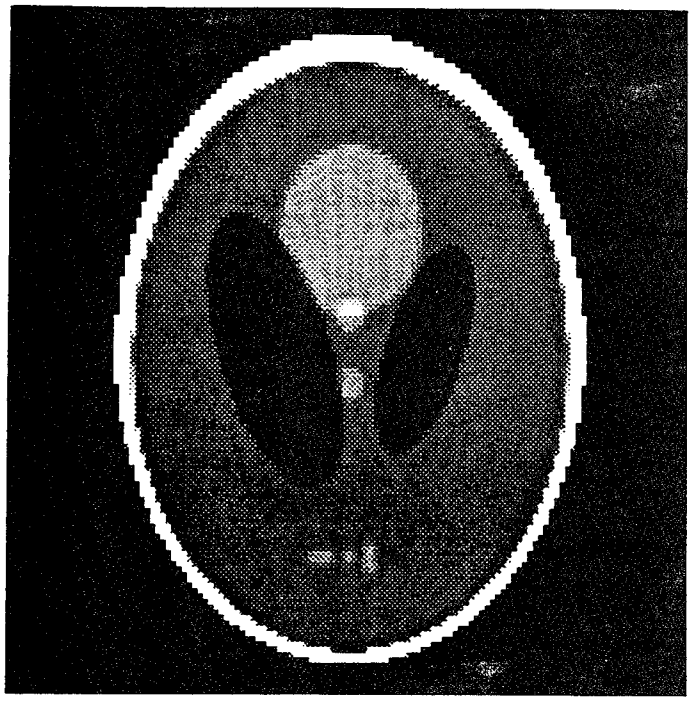


Figure 1.

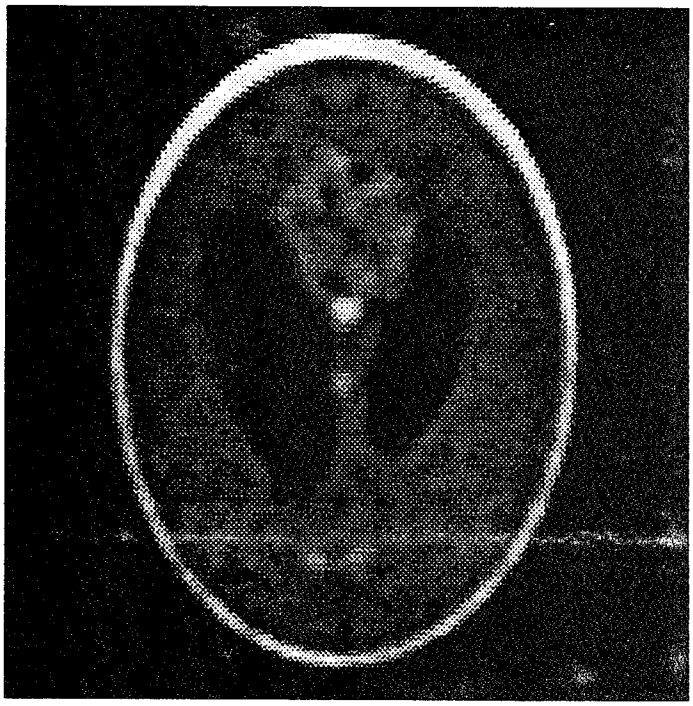


Figure 2.

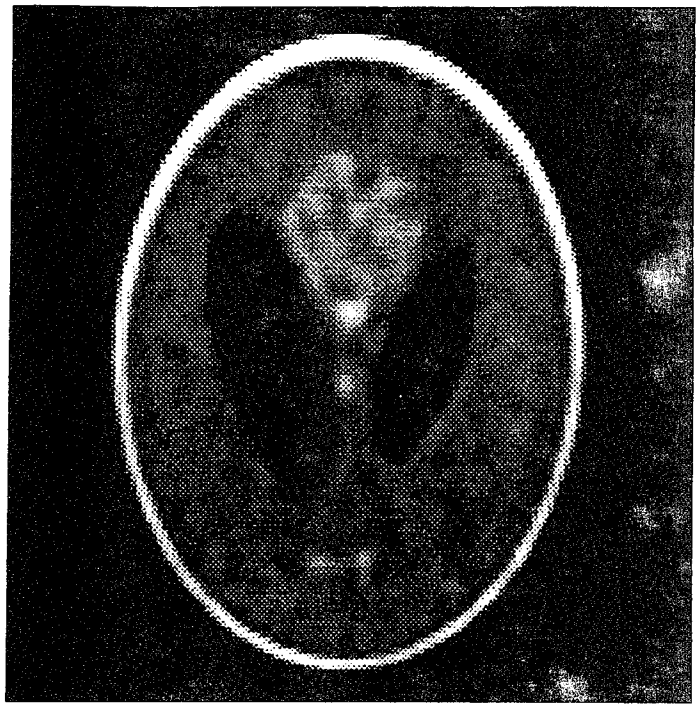


Figure 3.

Figure 4.

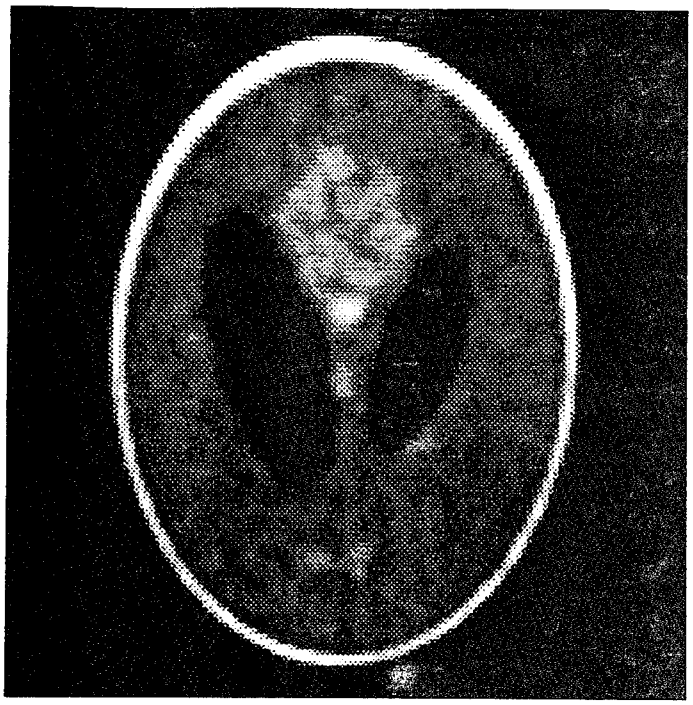


Figure 5.

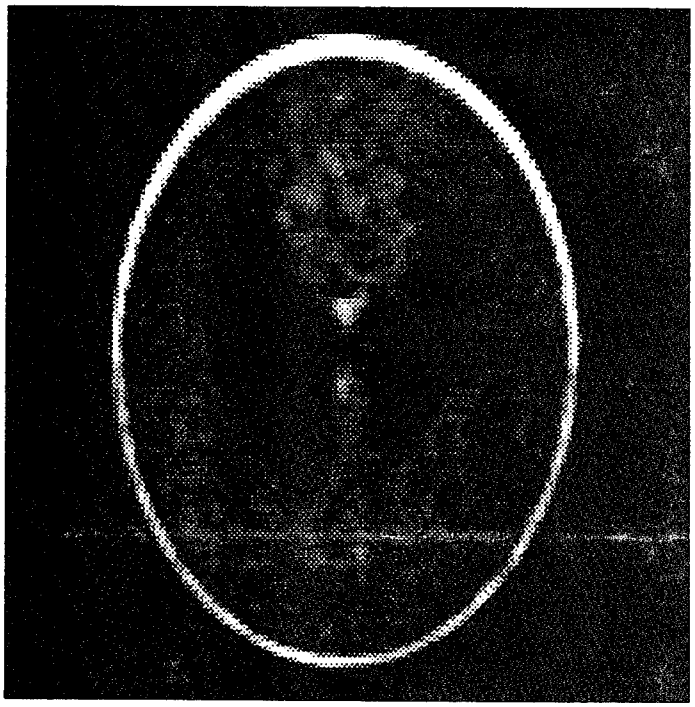


Figure 6.

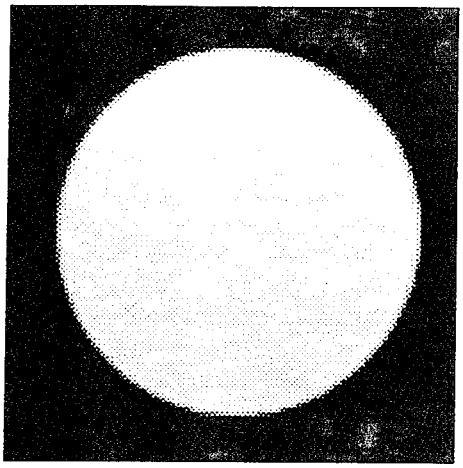


Figure 7.

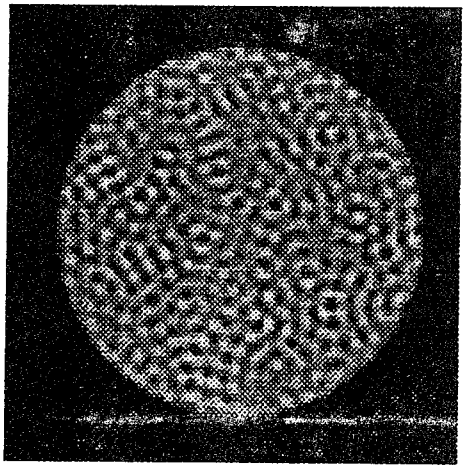


Figure 8.

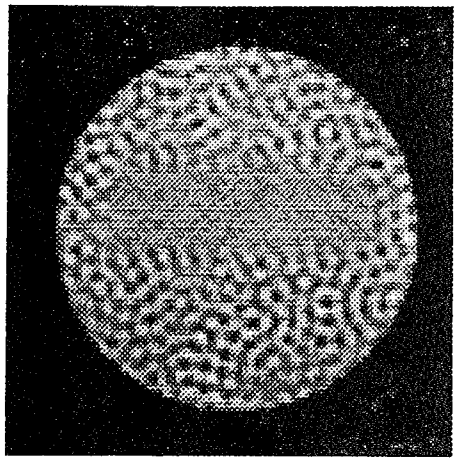


Figure 9.

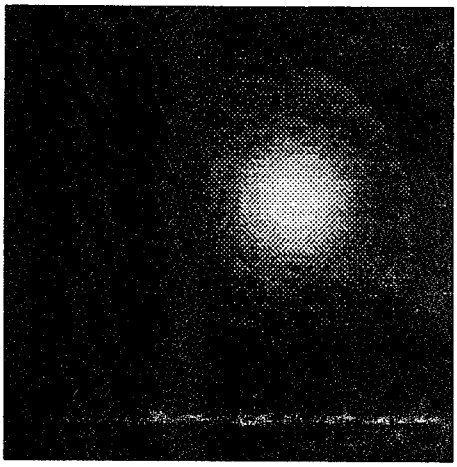


Figure 10.

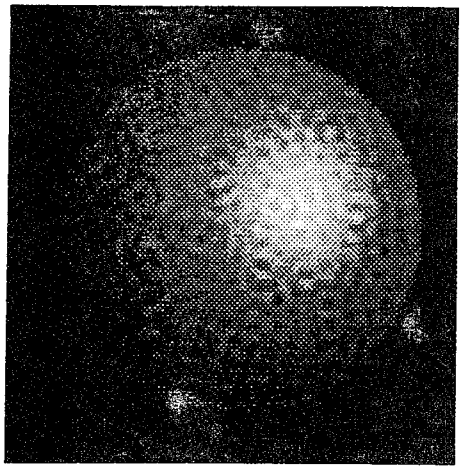


Figure 11.

Figure 12.

Figure 13.

Figure 14.

APPENDIX P

H. Soltanian-Zadeh, J.P. Windham, D.J. Peck, and A.E. Yagle, "A Comparative Analysis of Several Transformations for Enhancement and Segmentation of Magnetic Resonance Image Scene Sequences," IEEE Trans. Medical Imaging 11(3), 302-318, September 1992.

Its title describes this paper very well.

A Comparative Analysis of Several Transformations for Enhancement and Segmentation of Magnetic Resonance Image Scene Sequences

Hamid Soltanian-Zadeh, Joe P. Windham, Donald J. Peck, and Andrew E. Yagle, *Member, IEEE*

Abstract— **We compare the performance of the eigenimage filter to that of several other filters, applied to magnetic resonance image (MRI) scene sequences for images enhancement and segmentation. Comparisons are made with principal component analysis, matched, modified-matched, maximum contrast, target point, ratio, log-ratio, and angle image filters. Signal-to-noise ratio (SNR), contrast-to-noise ratio (CNR), segmentation of a desired feature (SDF), and correction for partial volume averaging effects (CPV) are used as performance measures. For comparison, analytical expressions for SNRs and CNRs of filtered images are derived, and CPV by a linear filter is studied. Properties of filters are illustrated through their applications to simulated and acquired MRI sequences of a phantom study and a clinical case; advantages and weaknesses are discussed. Our conclusion is that the eigenimage filter is the optimal linear filter that achieves SDF and CPV simultaneously.**

I. Introduction

CONTRAST in MR images depends on at least five major intrinsic tissue parameters: proton density (N(H)); spin-lattice (T1) and spin-spin (T2) relaxation times; flow velocity (ν); and chemical shift (δ). It also depends on four parameters of the pulse sequence: repetition time (TR); echo time (TE); inversion time (TI); and flip angle (θ). Hence, images are often difficult to interpret, and the observer must extract relevant information from all images in the MRI scene sequence. A variety of filters have been developed to proceed with this difficult task of picture interpretation. A transformation (filtration) can be viewed as an information rearrangement of data, so that the information in the transformed domain is easier to visualize and interpret. Some transformations achieve data reduction by removing extraneous information, such as interfering features or noise. Hence, they can be used for contrast enhancement and feature extraction, as well as for data compression. We proceed with a brief review of some well-known transformations applicable to MRI scene sequences (some details are given in Sections V-A, -B, and -C, in the Appendix).

Manuscript received April 1, 1991; revised January 14, 1992. This work was supported in part by NIH under Grant RO1 CA461240.

H. Soltanian-Zadeh is with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109, and the Department of Diagnostic Radiology and Medical Imaging, Henry Ford Hospital, Detroit, MI 48202.

J. P. Windham and D. J. Peck are with the Department of Diagnostic Radiology and Medical Imaging, Henry Ford Hospital, Detroit, MI 48202.

A. E. Yagle is with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109.

IEEE Log Number 9200404.

A. Review of Linear Transformations

A linear transformation which has been applied in a variety of fields [1]–[16], including MRI [14]–[16], is *principal component analysis* (PCA). PCA is a method of statistical analysis which has been employed in digital image processing as a technique for image coding, compression, enhancement, and feature extraction [6]–[9]. This filter gives linear combinations of the images which maximize the variance over a region of interest (ROI). The process also reduces the dimensionality of the useful data space. The first PCA image is the linear combination that achieves maximum global signal-to-noise ratio (GSNR) [17].

Sometimes, maximizing the signal-to-noise ratio (SNR) of only a *particular* feature is of interest. In this situation, a *matched filter* is optimum. The matched fiber was originally derived as the optimal linear filter for maximizing the output SNR [18]. It has also been used as the optimum receiver for detection of a known signal in white noise [19], and for medical image enhancement [20]. The basic matched filter does not remove any features from the scene. Its modified version (*modified-matched filter*) [21]–[23] removes any *constant* (*bias*) feature, at the expense of a decrease in SNR of the desired feature as compared to the basic matched filter.

In many applications, there are non-constant features which interfere with the observation of the desired object. To remove these interfering features at the lowest possible cost (least decrease in SNR of the desired feature), the *eigenimage filter* has been developed [24]–[28]. The eigenimage filter was originally derived as the linear filter which maximizes the ratio of a desired feature energy to one or more undesired (interfering) features energies in a composite image called the *eigenimage* [24]–[26]. We have recently found that this filter maximizes the SNR of the projection of a desired feature while suppressing the projections of interfering features in the eigenimage. Moreover, as we prove in Section V-E in the Appendix, it has the advantage of correcting for partial volume averaging effects (CPV).

In some applications, maximization of the contrast-to-noise ratio (CNR) between two features is desirable. In this case, the *maximum contrast* (*difference*) *filter* is the optimum solution [29]. The maximum contrast filter was originally derived for detecting one of two known signals in white noise [19]. There is no unique extension of the procedure to the general case of multiple interfering features in the scene. One possibility

0278–0062/92$03.00 © 1992 IEEE

is to use PCA. Another possibility is to use the linear filter which provides the largest value for the minimum absolute CNR (max. min. |CNR|) between a desired feature and multiple interfering processes [30]. This requires a search among several possibilities to find the optimal filter; we refer to this filter as *maximized minimum absolute* CNR (MMAC).

B. Review of Nonlinear Transformations

The nonlinear transformation which maximizes the CNR between an ROI and multiple interfering features is the *target point image method* (TPIM) [31]. In this approach, the filtered image is formed by calculating the Euclidean distance between each pixel vector and a target signature vector which is modeled or estimated. TPIM is analogous to using a separate matched filter for each location-dependent contrast vector; this maximizes CNR between the target region and each pixel. In a target point image (TPI), the values assigned to the pixels in the target region are near zero, while all other pixels in the image have larger values. The intensities in a TPI can be inverted, so that the target region appears white instead of black, consistent with the output of a linear filter. We refer to this inverted image as an *inverted target point image* (ITPI).

The *ratio filter* is another nonlinear filter which has previously been used for multispectral image enhancement in the field of remote sensing [32]–[34]. In many multispectral imaging systems the image can be modeled by the product of an object reflectivity function and an illumination function which is almost identical for all multispectral images. Dividing two such images provides an automatic normalization or compensation of the illumination factor. The same idea is applicable to T2-weighted multiple spin-echo MR images. Here, the signal can be approximated by a random variable multiplied by a damping exponential (T2 decay). It is known that intrinsic parameters (N(H), T1, and T2) of a specific tissue have random distributions. In a ratio image, most of the signal variations within a specific tissue are compensated (see Section V-C in the Appendix for details). This can improve the SNR and CNR of the image. In general, the ratio image between any two images can be defined as long as the divisor image is everywhere nonzero. For an image sequence, the ratio images can also be computed with respect to an average image; this reduces the propagating noise.

A problem with the ratio filter is the accentuation of the gray scale noise associated with each image. This can be reduced significantly by homomorphic filtering, i.e., taking logarithms of the ratio images. This generates *log-ratio* images [32]–[34]. If there are very small gray levels in a ratio image, the logarithm function generates very large negative numbers, resulting in a very large dynamic range for the log-ratio image. This may be avoided by taking the arctangent of a ratio image; this produces an *angle image* [35].

Other nonlinear transformations generate *calculated* or *synthesized images* [36]–[45]. Proton density and relaxation time images can be calculated from a set of acquired images. It is also possible to synthesize a contrast-optimized image with optimal contrast between two tissue types. Due to the propagation of noise, however, the conspicuity of these images are usually no better than the best acquired image [46], [29], [14]. Moreover, their calculations involve solving nonlinear equations which are quite time consuming. Therefore, we do not consider them here.

In previous work little has been done to 1) *apply* the ratio, log ratio, and angle image filters to MRI images, 2) *derive* analytical expressions for the SNR and CNR of composite images (for most of the filters) and 3) *compare* their performances when applied to MRI scene sequences. Such a study would be helpful to those who want to choose the most appropriate filter for their particular needs.

C. Our Contribution

New contributions of this work are five-fold: 1) application of ratio, log-ratio, and angle image filters to MRI (see Section V-C in the Appendix for a rationale of this application); 2) specification of analytical expressions for SNR's and CNR's; 3) evaluation of these expressions by comparing computed and actual SNR's and CNR's for all of the above filters; 4) investigation of CPV by a linear filter; and 5) illustration of the filters' properties using simulated and acquired MR images.

In Section II, we briefly discuss our approach and criteria for comparing composite images. In Section III, we apply the above filters to simulated and acquired MRI sequences of an egg phantom and a human brain. In Section IV, we compare the results and discuss the advantages and weaknesses of each transformation for enhancement, segmentation, and partial volume correction of MRI scene sequences. Appendix includes: some details of filters; rationale for the application of ratio filter to MRI; methods of deriving SNR and CNR expressions; proof of correcting for partial volume averaging affects by the eigenimage filter; list of abbreviations; and list of notations.

II. Performance Measures

In this study, we use simulated and acquired MR images of an egg phantom and a human brain. Each sequence consists of four T2-weighted and one T1-weighted MR images, for access to proton density, T1, and T2 information using only two acquisitions in a clinically-reasonable amount of time. We use a multiple spin-echo to generate four T2-weighted images, and a single spin-echo to generate a T1-weighted image. For performance evaluation, we consider SNR, CNR, SDF, and CPV in composite images generated by the above filters.

A. Signal-to-Noise Ratio

For image interpretation, one is usually interested in visualization of a specific tissue referred to as the desired feature. Let P^d_{jk} be the gray level of the (j,k)th pixel in the desired ROI (DROI). P^d_{jk} consists of a deterministic value plus statistical noise; the desired deterministic value if then the mean $E[P^d_{jk}]$, and the strength of the noise is the standard deviation $(\mathrm{Var}(P^d_{jk}))^{1/2}$. Signal-to-noise ratio of the desired

feature (SNR_d) is defined as

$$\mathrm{SNR}_d = \frac{E\left[P_{jk}^d\right]}{\sqrt{\mathrm{Var}\left(P_{jk}^d\right)}}. \tag{1}$$

Since statistical noise in MRI is ergodic and uncorrelated with the signal [47]–[48], $E[P_{jk}^d]$ and $\mathrm{Var}(P_{jk}^d)$ in a homogeneous region of M pixels can be estimated using the sample mean and variance [49]

$$\hat{E}\left[P_{jk}^d\right] = \frac{1}{M}\sum_{j,k} P_{jk}^d \tag{2}$$

$$\mathrm{Var}\left(P_{jk}^d\right) = \frac{1}{M-1}\sum_{j,k}\left(P_{jk}^d - \frac{1}{M}\sum_{j,k} P_{jk}^d\right)^2. \tag{3}$$

The estimated values $\hat{E}[P_{jk}^d]$ and $\hat{\mathrm{Var}}(P_{jk}^d)$ are then inserted into (1). We make the following assumptions which were also made in [23], [29], [30], [31]: 1) statistical noise in MRI can be modeled as a Gaussian distributed zero-mean white noise field with standard deviation σ; and 2) signature vectors are *a priori* known fairly well. Then, the standard formula for noise propagation [50], [51] ((23)–(24) in Section V-D in the Appendix) shows that SNR of the desired object in a composite image generated by a linear filter is given by

$$\mathrm{SNR}_d = \frac{\vec{W}\cdot\vec{d}}{\sigma\left(\vec{W}\cdot\vec{W}\right)^{\frac{1}{2}}} \tag{4}$$

where $\vec{W}$ and $\vec{d}$ are the weighting vector and the desired signature vector, respectively.

Using (4) and the analytical expressions for the weighting vectors (given in Section V-A in the Appendix), we have derived the mathematical expressions for SNR of the desired feature in linearly filtered images. For TPI's and ITPI's, we used the fact that the squared Euclidean distance between a pixel vector in the DROI and the desired signature vector has a chi-squared distribution with n degrees of freedom where n is the number of images in the sequence. Hence, the expected values and standard deviations necessary for SNR_d's can be determined (see Section V-D in the Appendix for details). For ratio, log-ratio, and angle images, we have derived mathematical expressions for SNR_d's in the corresponding composite images using the standard formula for noise propagation (23)–(24). The resulting analytical expressions for all of these filters are summarized in the second columns of Tables I and II; details of the deriviations are given in [52].

B. Contrast-to-Noise Ratio

In addition to SNR_d, the CNR between the desired tissue and an undesired (interfering) feature (background or another tissue type) is usually important for image interpretation. The CNR between the desired feature and an undesired feature (CNR_{du}) is defined as [29]

$$\mathrm{CNR}_{du} = \frac{E\left[P_{jk}^d\right] - E\left[P_{jk}^u\right]}{\sqrt{\frac{\mathrm{Var}(P_{jk}^d)+\mathrm{Var}(P_{jk}^u)}{2}}} \tag{5}$$

where P_{jk}^u is the gray level of the (j,k)th pixel in the undesired ROI (UROI), and $E[P_{jk}^u]$ and $\mathrm{Var}(P_{jk}^u)$ are the mean and the variance of pixel values in the UROI, respectively.

Making the assumptions and using the techniques described for SNR_d, we have derived the CNR_{du} for the above filters. The resulting expressions are summarized in the third and fourth columns of Table I and the third column of Table II; details of the derivations are given in [52].

C. Segmentation of a Desired Feature

If a filtered image can be windowed (linearly histogram equalized) so that only the desired feature is visualized in the scene, the corresponding filter has *segmented* the desired feature (SDF). Although SNR_d and CNR_{du} give quantitative measures of the image quality, they do not note whether SDF from undesired features is achieved. Therefore, we consider SDF in evaluating the composite image quality; complete SDF is achieved if and only if all interfering feature signature vectors are mapped into gray levels which are *all* less than or *all* greater than the gray level into which the desired signature vector is mapped. In this paper, we evaluate SDF by inspection.

D. Correction for Partial Volume Averaging Effects

We assume that the magnetic resonance signal S from a voxel containing m different materials is given by

$$S = \sum_{l=1}^{m}\left(\frac{V_l}{V}\right)S_l \tag{6}$$

where V_l is the volume of the lth material within the voxel, V is the total volume of the voxel, and S_l is the signal from the lth material. This is a reasonable assumption, since the signal from a voxel is directly proportional to the net macroscopic magnetization. The net macroscopic magnetization is the sum of all of the individual magnetic moments, provided that the frequency band-width across the voxel is larger than the chemical shifts of the different chemical materials in the voxel. Hence, the gray level P_{jk} of the (j,k)th pixel (corresponding to the (j,k)th voxel) in an MR image is given by

$$P_{jk} = E[P_{jk}] + \eta_{jk} = \sum_{l=1}^{m}\left(\frac{V_{ljk}}{V}\right)S_l + \eta_{jk} \tag{7}$$

where V_{ljk} is the partial volume of the lth material in the (j,k)th voxel, and η_{jk} represents statistical noise which is again assumed to be an additive zero-mean white Gaussian noise field with standard deviation σ, uncorrelated with the signal.

Correction for partial volume averaging effects (CPV) means that in the transformed image (TI) we obtain

$$E[TI_{jk}] = \left(\frac{V_{djk}}{V}\right)E\left[T\left(P_{lm}^d\right)\right] \tag{8}$$

TABLE I
ANALYTICAL EXPRESSIONS FOR SNR_d, CNR_{du_1}, AND CNR_{du_2} OF LINEAR FILTERS (THREE FEATURES IN THE SCENE)

Filter	SNR_d	CNR_{du_1}	CNR_{du_2}
PCA_1	$\frac{\|\vec{d}\|}{\sigma}\cos(\beta)$	$\frac{\|\vec{d}-\vec{u_1}\|}{\sigma}\cos(\beta_1)$	$\frac{\|\vec{d}-\vec{u_2}\|}{\sigma}\cos(\beta_2)$
Matched	$\frac{\|\vec{d}\|}{\sigma}$	$\frac{\|\vec{d}\|}{\sigma}-\frac{\|\vec{u_1}\|}{\sigma}\cos(\theta_1)$	$\frac{\|\vec{d}\|}{\sigma}-\frac{\|\vec{u_2}\|}{\sigma}\cos(\theta_2)$
Mod-Mat	$\frac{\|\vec{d}\|}{\sigma}\sin(\phi)$	A1	A2
Eigen	$\frac{\|\vec{d}\|}{\sigma}\sin(\theta)$	$\frac{\|\vec{d}\|}{\sigma}\sin(\theta)$	$\frac{\|\vec{d}\|}{\sigma}\sin(\theta)$
$MMAC_1$	$\frac{\|\vec{d}\|}{\sigma}\cos(\gamma_1)$	$\frac{\|\vec{d}-\vec{u_1}\|}{\sigma}$	$\frac{\|\vec{d}-\vec{u_2}\|}{\sigma}\cos(\gamma_3)$
$MMAC_2$	$\frac{\|\vec{d}\|}{\sigma}\cos(\gamma_2)$	$\frac{\|\vec{d}-\vec{u_1}\|}{\sigma}\cos(\gamma_3)$	$\frac{\|\vec{d}-\vec{u_2}\|}{\sigma}$
$MMAC_3$	A3	$\frac{\|\vec{d}-\vec{u_1}\|\|\vec{d}-\vec{u_2}\|}{\sigma\|\vec{u_1}-\vec{u_2}\|}\sin(\gamma_3)$	CNR_{du_1}
$MMAC_4$	A4	$-\frac{\|\vec{d}-\vec{u_1}\|\|\vec{d}-\vec{u_2}\|}{\sigma\|2\vec{d}-\vec{u_1}-\vec{u_2}\|}\sin(\gamma_3)$	$-CNR_{du_1}$

$$A1 = \frac{\|\vec{d}\|}{\sigma}\sin(\phi) - \frac{\|\vec{u_1}\|}{\sigma}\frac{\cos(\theta_1)-\cos(\phi)\cos(\alpha_1)}{\sin(\phi)}$$

$$A2 = \frac{\|\vec{d}\|}{\sigma}\sin(\phi) - \frac{\|\vec{u_2}\|}{\sigma}\frac{\cos(\theta_2)-\cos(\phi)\cos(\alpha_2)}{\sin(\phi)}$$

$$A3 = \frac{\|\vec{d}\|}{\sigma}\frac{\|\vec{d}-\vec{u_1}\|[\cos(\gamma_2)-\cos(\gamma_1)\cos(\gamma_3)]+\|\vec{d}-\vec{u_2}\|[\cos(\gamma_1)-\cos(\gamma_2)\cos(\gamma_3)]}{\|\vec{u_1}-\vec{u_2}\|\sin(\gamma_3)}$$

$$A4 = \frac{\|\vec{d}\|}{\sigma}\frac{\|\vec{d}-\vec{u_1}\|[\cos(\gamma_2)-\cos(\gamma_1)\cos(\gamma_3)]-\|\vec{d}-\vec{u_2}\|[\cos(\gamma_1)-\cos(\gamma_2)\cos(\gamma_3)]}{\|2\vec{d}-\vec{u_1}-\vec{u_2}\|\sin(\gamma_3)}$$

where $E[TI_{jk}]$ is the mean value of the (j,k)th pixel in the transformed image, V_{djk} is the partial volume of the desired material in the (j,k)th voxel, and $E[T(P^d_{lm})]$ is the mean value of the (l,m)th pixel in the desired ROI of the transformed image. This correction is necessary for correct interpretation and analysis of MR images, as well as for volume calculations. Since none of the previous factors (SNR_d, CNR_du, and SDF) note whether this correction is achieved, we consider it as a fourth factor in the evaluation of composite image quality. In Section V-E in the Appendix we show that the eigenimage filter achieves CPV, and that none of the other linear filters listed above does so.

III. SIMULATION AND ACQUIRED IMAGE RESULTS

A. Discussion of Analytical Expressions

For a scene with three features, we have derived and listed the SNR'_ds and CNR'_{du}s of linear filters in Table I. The SNR_d of the matched filter is the highest in the list, since all other SNR'_ds have sinusoidal factors. The CNR'_{du}s of the MMAC filter are the highest among all CNR'_{du}s in the list. The eigenimage filter is unique in that it has equal SNR_d and CNR'_{du}s.

Analytical expressions for the SNR'_ds and CNR'_{du}s of nonlinear filters are listed in Table II. Note that these expressions can be used for a scene with arbitrary number of interfering features. Except for the ratio, log-ratio, and angle images (computed using only two images in the sequences), all other SNR's and CNR's monotonically increase with the number of images in the sequence. As expected, SNR's and CNR's of ratio, log-ratio, and angle images calculated with respect to the average image are larger than the corresponding ratios for composite images calculated using only two images in the sequence.

The analytical expressions in Tables I and II can be used to estimate the SNR'_ds and CNR'_{du}s of the filtered images without actual filtering of the image sequence. In addition to the comparison, this provides the possibility of optimizing MRI pulse sequence parameters for each filter; we discuss such an optimization procedure for the eigenimage filter in a companion paper [53].

TABLE II
ANALYTICAL EXPRESSIONS FOR SNR_d AND CNR_{du} OF NONLINEAR FILTERS (MULTIPLE FEATURES IN THE SCENE)

Filter	SNR_d	CNR_{du}
TPI	$\frac{1}{\sqrt{\frac{n}{2}\left(\frac{\Gamma(n/2)}{\Gamma((n-1)/2)}\right)^2-1}}$	$\frac{\frac{\sqrt{2}\,\Gamma((n-1)/2)}{\Gamma(n/2)}-\frac{\lVert\vec{d}-\vec{u}\rVert}{\sigma}}{\sqrt{\frac{n-1}{2}-\left(\frac{\Gamma((n-1)/2)}{\Gamma(n/2)}\right)^2}}$
ITPI	$\frac{1}{\sqrt{\frac{2}{n-2}\left(\frac{\Gamma(n/2)}{\Gamma((n-1)/2)}\right)^2-1}}$	$\frac{\frac{\Gamma((n-1)/2)}{\Gamma(n/2)}-\frac{\sigma\sqrt{2}}{\lVert\vec{d}-\vec{u}\rVert}}{\sqrt{\frac{1}{n-2}-\frac{1}{2}\left(\frac{\Gamma((n-1)/2)}{\Gamma(n/2)}\right)^2+\left(\frac{\sigma}{\lVert\vec{d}-\vec{u}\rVert}\right)^4}}$
R^{lm}	$\frac{d_l}{\sigma\sqrt{1+\left(\frac{d_l}{d_m}\right)^2}}$	$\frac{\sqrt{2}}{\sigma}\,\frac{d_l u_m-u_l d_m}{\sqrt{u_m^2\left(1+\left(\frac{d_l}{d_m}\right)^2\right)+d_m^2\left(1+\left(\frac{u_l}{u_m}\right)^2\right)}}$
LR^{lm}	$\frac{\ln(d_l)-\ln(d_m)}{\sigma\sqrt{\left(\frac{1}{d_l}\right)^2+\left(\frac{1}{d_m}\right)^2}}$	$\frac{\sqrt{2}}{\sigma}\,\frac{\ln(d_l u_m)-\ln(u_l d_m)}{\sqrt{\left(\frac{1}{d_l}\right)^2+\left(\frac{1}{u_l}\right)^2+\left(\frac{1}{d_m}\right)^2+\left(\frac{1}{u_m}\right)^2}}$
AG^{lm}	$\frac{1}{\sigma}\left[\arctan\left(\frac{d_l}{d_m}\right)\right]\sqrt{d_l^2+d_m^2}$	$\frac{\sqrt{2}}{\sigma}\,\frac{\arctan\left(\frac{d_l}{d_m}\right)-\arctan\left(\frac{u_l}{u_m}\right)}{\sqrt{\frac{1}{d_l^2+d_m^2}+\frac{1}{u_l^2+u_m^2}}}$
R^{la}	$\frac{d_l}{\sigma\sqrt{1+\frac{1}{n}\left(\frac{d_l}{d_a}\right)^2-\frac{2}{n}\frac{d_l}{d_a}}}$	$\frac{\sqrt{2}}{\sigma}\,\frac{d_l u_a-u_l d_a}{\sqrt{u_a^2\left[1+\frac{1}{n}\left(\frac{d_l}{d_a}\right)^2-\frac{2}{n}\frac{d_l}{d_a}\right]+d_a^2\left[1+\frac{1}{n}\left(\frac{u_l}{u_a}\right)^2-\frac{2}{n}\frac{u_l}{u_a}\right]}}$
LR^{la}	$\frac{\ln(d_l)-\ln(d_a)}{\sigma\sqrt{\left(\frac{1}{d_l}\right)^2+\frac{1}{n}\left(\frac{1}{d_a}\right)^2-\frac{2}{n}\frac{1}{d_l d_a}}}$	$\frac{\sqrt{2}}{\sigma}\,\frac{\ln(d_l u_a)-\ln(u_l d_a)}{\sqrt{\left(\frac{1}{d_l}\right)^2+\left(\frac{1}{u_l}\right)^2+\frac{1}{n}\left[\left(\frac{1}{d_a}\right)^2+\left(\frac{1}{u_a}\right)^2\right]-\frac{2}{n}\left[\frac{1}{d_l d_a}+\frac{1}{u_l u_a}\right]}}$
AG^{la}	$\frac{\left[\arctan\left(\frac{d_l}{d_a}\right)\right](d_l^2+d_a^2)}{\sigma\sqrt{d_a^2+\frac{1}{n}d_l^2-\frac{2}{n}d_a d_l}}$	$\frac{\sqrt{2}}{\sigma}\,\frac{\arctan\left(\frac{d_l}{d_a}\right)-\arctan\left(\frac{u_l}{u_a}\right)}{\sqrt{\frac{d_a^2+\frac{1}{n}d_l^2-\frac{2}{n}d_a d_l}{(d_l^2+d_a^2)^2}+\frac{u_a^2+\frac{1}{n}u_l^2-\frac{2}{n}u_a u_l}{(u_l^2+u_a^2)^2}}}$

Notations used in Tables I and II

α_1: The angle between $\vec{u_1}$ and the constant vector $\vec{c}=[1, 1, \cdots, 1]^T$.

α_2: The angel between $\vec{u_2}$ and the constant vector $\vec{c}$.

β: The angle between $\vec{d}$ and the eigenvector corresponding to the maximum eigenvalue of the sample covariance matrix ($\vec{e}_m$).

β_1: The angle between $\left(\vec{d}-\vec{u}_1\right)$ and $\vec{e}_m$.

β_2: The angle between $\left(\vec{d}-\vec{u}_2\right)$ and $\vec{e}_m$.

θ: The angle between $\vec{d}$ and its projection onto the subspace spanned by $\vec{u}_1$ and $\vec{u}_2$.

θ_1: The angle between $\vec{d}$ and $\vec{u}_1$.

θ_2: The angle between $\vec{d}$ and $\vec{u}_2$.

ϕ: The angle between $\vec{d}$ and the constant vector $\vec{c}$.

γ_1: The angle between $\vec{d}$ and $\left(\vec{d}-\vec{u}_1\right)$.

γ_2: The angle between $\vec{d}$ and $\left(\vec{d}-\vec{u}_2\right)$.

γ_3: The angle between $\left(\vec{d}-\vec{u}_1\right)$ and $\left(\vec{d}-\vec{u}_2\right)$.

$\Gamma(\cdot)$: The gamma function, $\Gamma(\delta)=\int_0^\infty x^{\delta-1}e^{-x}dx$, $\delta>0$.

n: The number of images in the original MRI scene sequence.

d_l: The average gray level in the DROI of the lth original image (lth element of $\vec{d}$).

d_a: The average gray level in the DROI of the average image (average of elements of $\vec{d}$).

u_l: The average gray level in the UROI of the lth original image (lth element of $\vec{u}$).

u_a: The average gray level in the UROI of the average image (average of elements of $\vec{u}$).

B. Specifications of Experiments

In order to illustrate the performance of different transformations applicable to MR scene sequences, we have used simulated and acquired MRI sequences. Each sequence consists of four T2-weighted and one T1-weighted MR images. Simulations were performed on a Vicom image processing computer. Acquired MRI sequences of an egg phantom, an agarose phantom, and a human brain were acquired on a 1.5 T General Electric Signa MRI system.

The first example is a simulated MRI sequence which contains three overlapping circular regions which simulates white matter, gray matter, and cerebrospinal fluid (CSF). The second and third examples are acquired MRI sequences of a shell-removed hard-boiled egg in gelatin, and a human brain, respectively. The fourth example is acquired MRI sequences of an agarose phantom to illustrate partial volume correction by the eigenimage filter in real life. For the first three examples, original image sequences are shown in Figs. 1, 4, and 7, respectively. Transformed images are shown in Figs. 2–3, 5–6, and 8–9, respectively. The original images in Figs. 1, 4, and 7 are windowed (linearly histogram equalized) together to provide optimal contrast for the observation of *all* features present in the scene. Transformed images are individually windowed to provide optimal contrast for the segmentation of our *desired* feature (egg-white for egg, white matter for the simulation and brain). For the agarose phantom, a spin-echo sagittal slice, four multiple spin-echo coronal slices, and two coronal eigenimages are shown in Fig. 11.

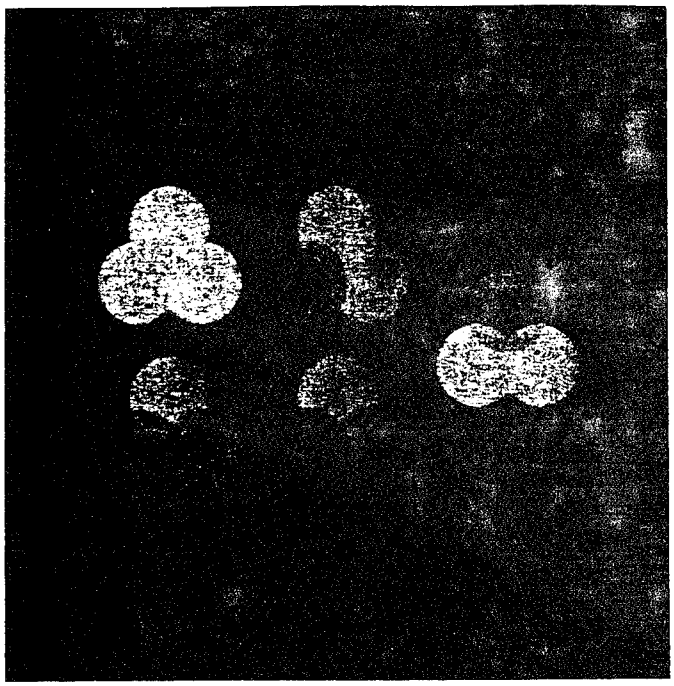


Fig. 1. (a)–(e) Four simulated spin-echo T2–weighted (TE/TR = 25–100/2500 ms) and one T1-weighted (TE/TR = 25/500 ms) MR images, respectively. The scene consists of three overlapping circular regions; the top circle represents CSF, the right circle represents grey matter, and the left circle represents white matter. Voxels corresponding to the pixels in the overlapping regions are assumed to contain equal proportions of the overlapping tissues. Images are windowed together to provide optimal contrast for the observation of all features in the scene.

C. Results for Linear and TPI Filters

The filters listed in Section I were implemented and tested on a Vicom image processing computer. Sample covariance matrices were estimated over the whole image pixels, and five PCA images were generated for each sequence. For the simulation, they contained 73.297%, 25.334%, 0.981%, 0.196%, and 0.192% of the variance, respectively. Note that the variance can be viewed as a measure of the image information content if it is computed over a significant portion of the image containing different tissues [15]. For the egg phantom, PCA images contained 95.364%, 4.466%, 0.135%, 0.018%, and 0.017% of the variance, respectively. For the brain, they contained 91.553%, 7.746%, 0.564%, 0.078%, and 0.059% of the variance, respectively.

The first and second PCA images are shown in Figs. 2(a), 5(a), 8(a), and 3(a), 6(a), 9(a), respectively. The second PCA images provides the best segmentation of our desired feature for the egg and brain examples, as compared to the other PCA images. This suggests that SDF is not necessarily achieved in the scene with the largest variance. Note that the first PCA image for the brain is quite similar to the matched filtered image for the white matter, since this is the dominant tissue type in the slice.

The matched image, eigenimage, first and second MMAC images, and target point image for the desired features are shown in Figs. 2(b)–(f), 5(b)–(f), and 8(b)–(f). The modified matched image, eigenimage, third and fourth maximum CNR images, and inverted target point image for the desired features are shown in Figs. 3(b)–(f), 6(b)–(f), and 9(b)–(f). Although the matched image provides the maximum

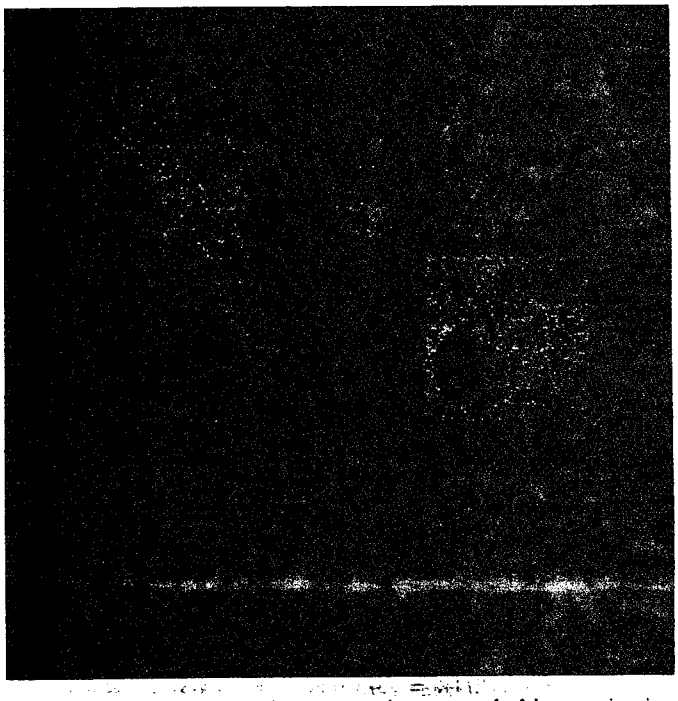


Fig. 2. (a)–(f) First principal component image, matched image, eigenimage, first and second MMAC images, and target point image for the left circle, respectively. (g)–(i) Ratio, log-ratio, and angle images obtained from dividing the T1-weighted image shown in Fig. 1(e) by the fourth T2-weighted image shown in Fig. 1(d). Images are windowed individually to provide optimal contrast for the segmentation of the left circle.

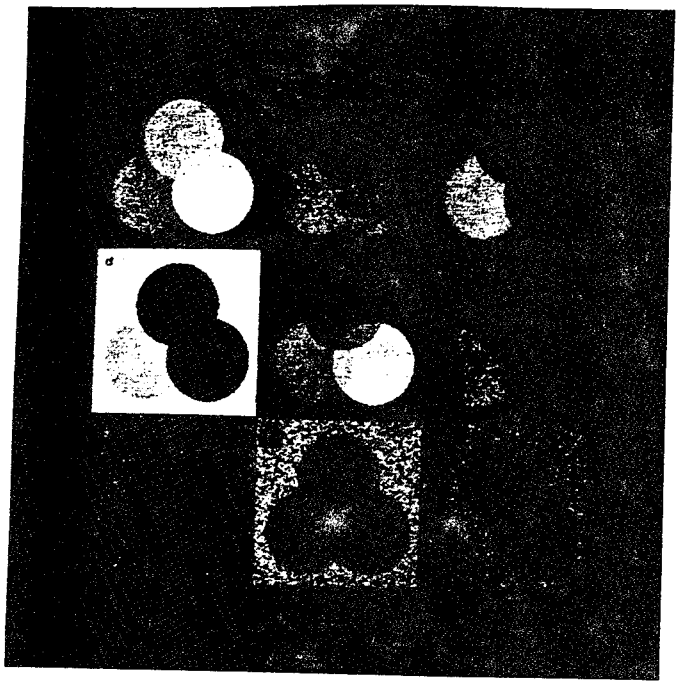


Fig. 3. (a)–(f) Second principal component image, modified-matched image, eigenimage, third and fourth MMAC images, and inverted target point image for the left circle, respectively. (g)–(i) Ratio, log-ratio, and angle images obtained from dividing the T1-weighted image shown in Fig. 1(e) by the average of all images shown in Fig. 1. Images are windowed individually to provide optimal contrast for the segmentation of the left circle.

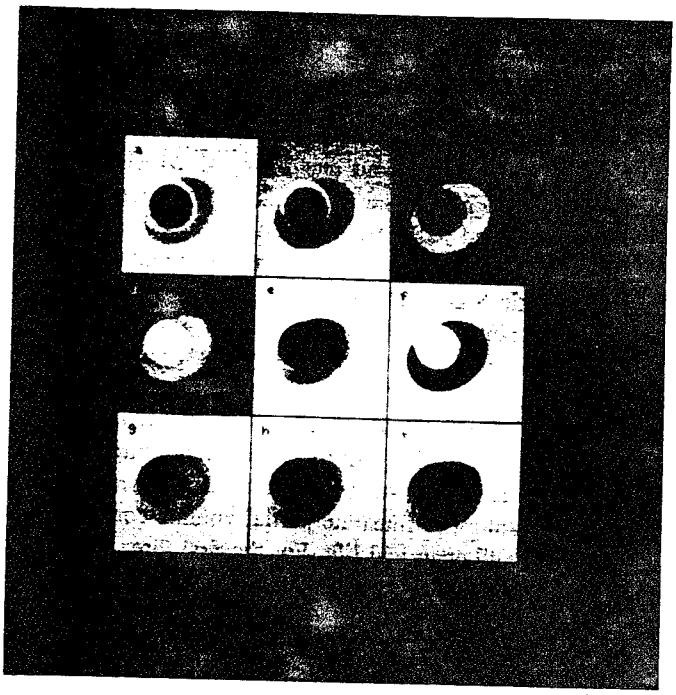


Fig. 5. (a)–(f) First principal component image, matched image, eigenimage, first and second MMAC images, and target point image for the egg-white, respectively. (g)–(i) Ratio, log-ratio, and angle images obtained from dividing the first T2-weighted image shown in Fig. 4(a) by the T1-weighted image shown in Fig. 4(e). Images are windowed individually to provide optimal contrast for the segmentation of the egg-white.

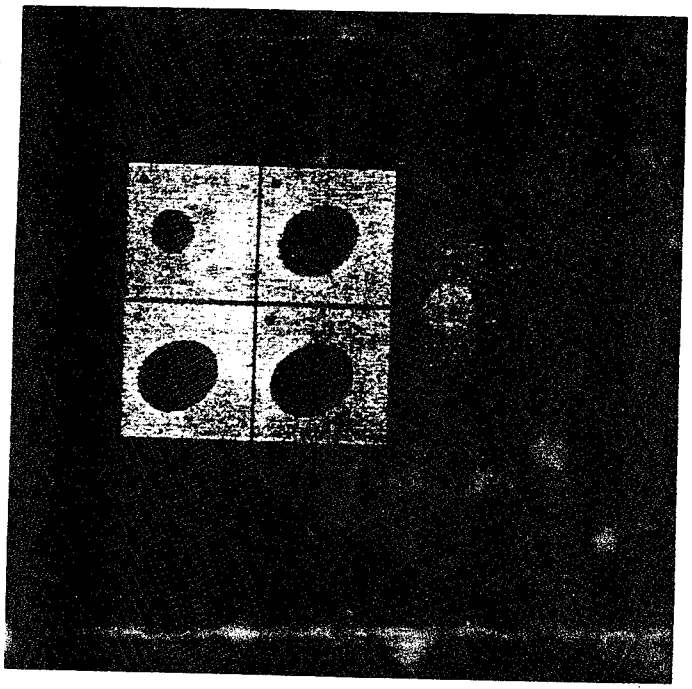


Fig. 4. (a)–(e) Four spin-echo T2-weighted (TE/TR = 25–2500 ms) and one T1-weighted (TE/TR = 25/500 ms) MR images of a shell-removed hard-boiled egg in gelatin, respectively. Images are windowed together to provide optimal contrast for the observation of all features in the scene. Note the zipper artifact in the T2-weighted images.

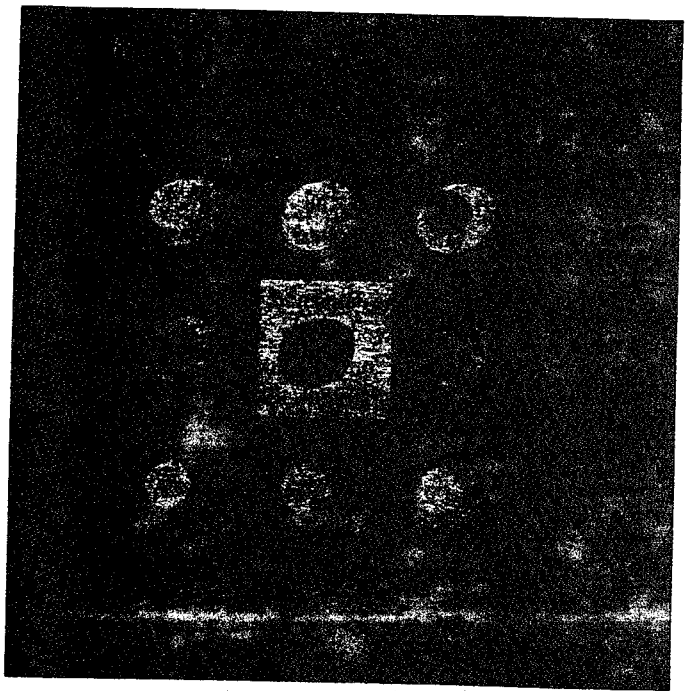


Fig. 6. (a)–(f) Second principal component image, modified-matched image, eigenimage, third and fourth MMAC images, and inverted target point image for the egg-white, respectively. (g)–(i) Ratio, log-ratio, and angle images obtained from dividing the T1-weighted image shown in Fig. 4(e) by the average of all images shown in Fig. 4. Images are windowed individually to provide optimal contrast for the segmentation of the egg-white.

possible SNR_d, its segmentation ability is inferior to that of the modified matched image, TPI, ITPI, and especially the eigenimage. This is due to the presence of interfering features in the matched image. The other composite images provide an inferior SNR_d but do partially remove undesired features from the scene. The windowed version of the third MMAC image shows the desired feature, while other features are

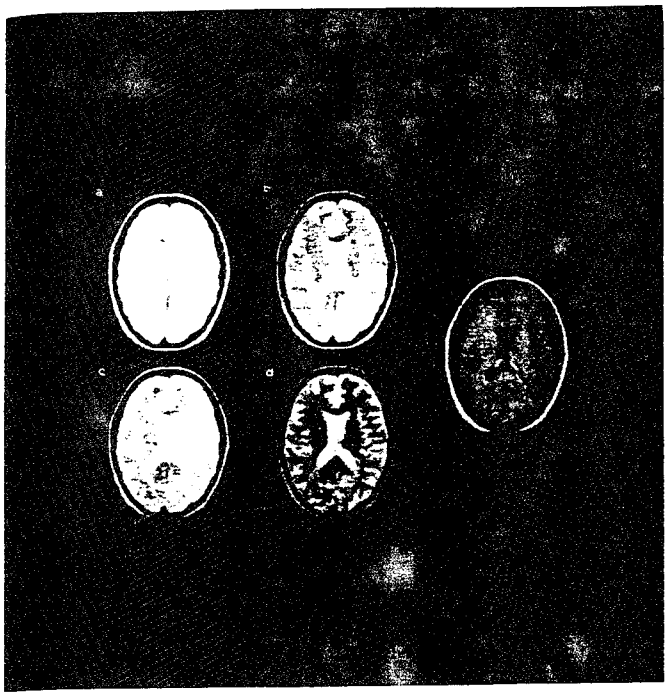


Fig. 7. (a)–(e) Four spin-echo T2-weighted (TE/TR = 25-100/2500 ms) and one T1-weighted (TE/TR = 25/500 ms) MR images of a human brain respectively. Images are windowed together to provide optimal contrast for the observation of all features in the scene.

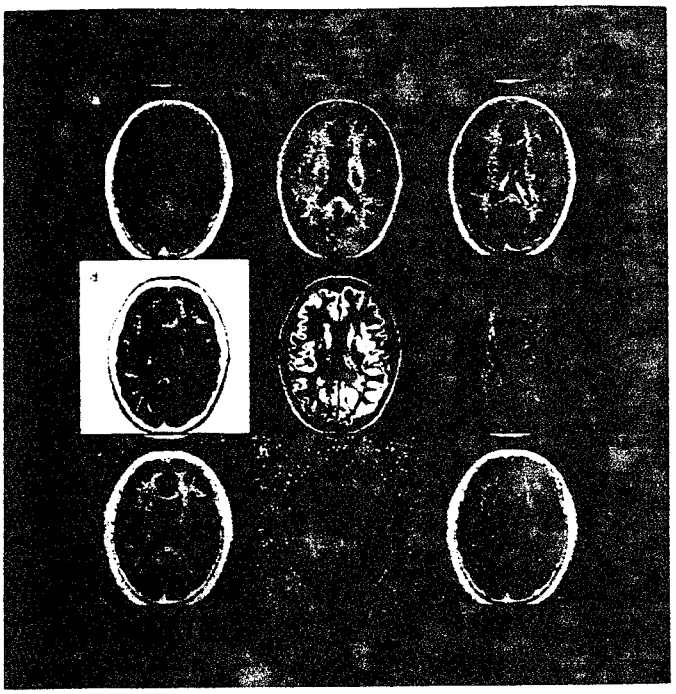


Fig. 9. (a)–(f) Second principal component image, modified-matched image, eigenimage, third and fourth MMAC images, and inverted target point image for the white matter, respectively. (g)–(i) Ratio, log-ratio, and angle images obtained from dividing the T1-weighted image shown in Fig. 7(e) by the average of all images shown in Fig. 7. Images are windowed individually to provide optimal contrast for the segmentation of the white matter.

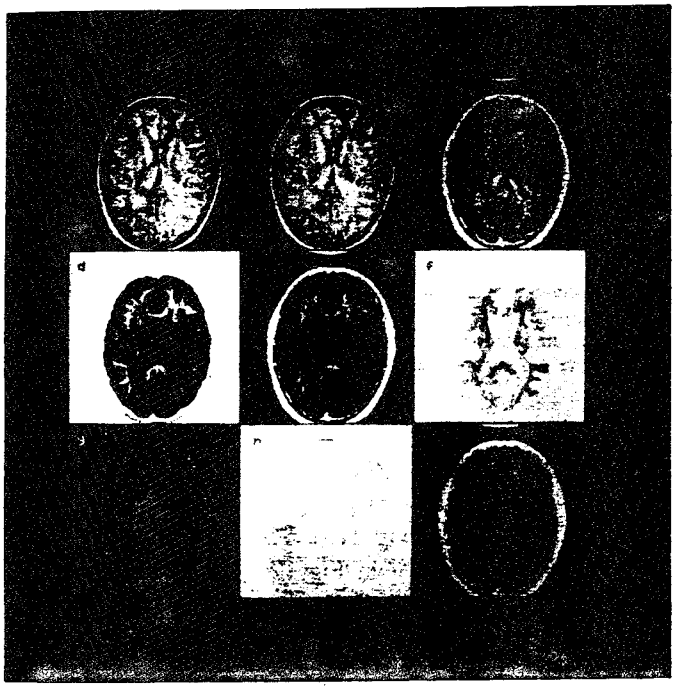


Fig. 8. (a)–(f) First principal component image, matched image, eigenimage, first and second MMAC images, and target point image for the white matter, respectively. (g)–(i) Ratio, log-ratio, and angle images obtained from dividing the T1-weighted image shown in Fig. 7(e) by the fourth T2-weighted image shown in Fig. 7(d). Images are windowed individually to provide optimal contrast for the segmentation of the white matter.

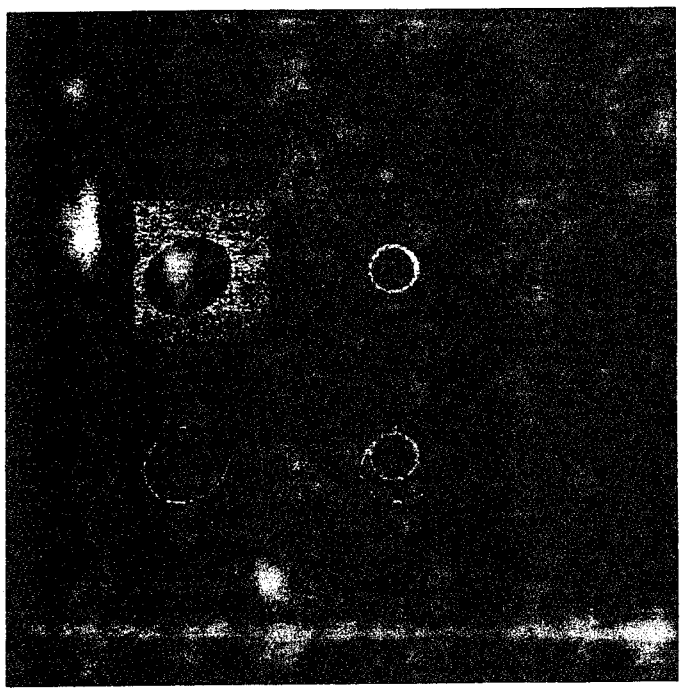


Fig. 10. (a) Third T2-weighted MR image of the egg phantom. (b)–(d) Maps of pixels corresponding to voxels containing partial volumes of egg-yolk and egg-white, egg-white and gelatin, and either of them, respectively.

removed. However, the partial volume averaging effects are not corrected.

Among all of these filters, the eigenimage filter removes interfering features *completely*, and also provides CPV. This correction is shown in the overlapping regions of the simulation in Fig. 2(c), between white and gray matters in Fig. 8(c), at the border of egg-white and egg-yolk in Fig. 5(c), and in the overlapping region of agarose phantom in Fig. 11(f)–(g). A map of pixels containing partial volume averaging effects for

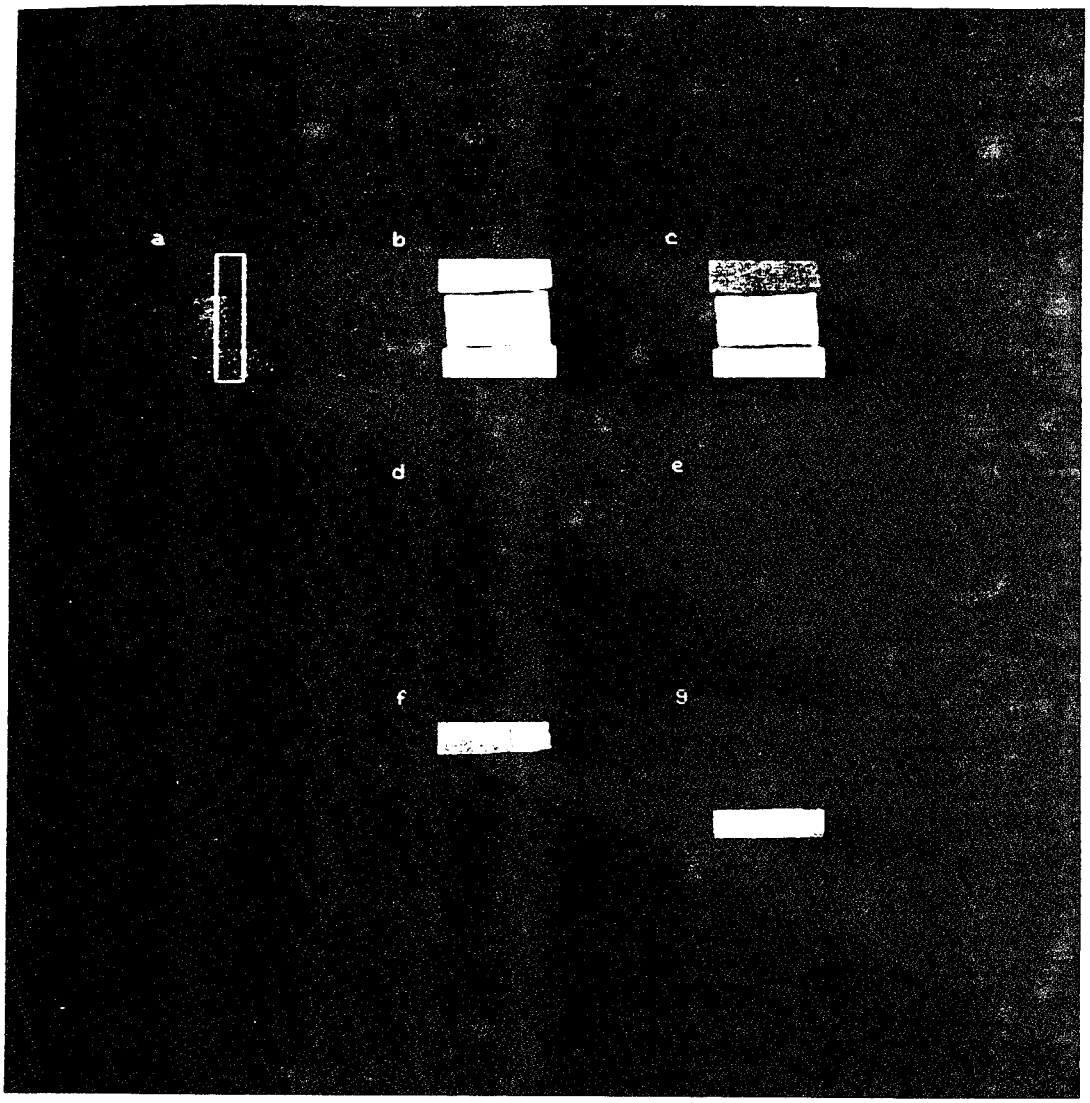


Fig. 11. (a) A sagittal view of an agarose phantom. Slice position for the coronal images in (b)–(e) is also shown. Dark and bright regions correspond to 3% and 2% agarose compounds, respectively. (b)–(e) Four coronal spin-echo T2-weighted (TE/TR = 25-100/2500 ms) MR images of the agarose phantom. Images in (b)–(e) are windowed together to provide optimal contrast for the observation of all features in the scene. (f)–(g) Eigenimages for the top and bottom regions, respectively. Note CPV for the central region in both cases.

the egg phantom is shown in Fig. 10. Estimated and original partial volumes for the simulation and the agarose phantom are given in Table IV. Regarding the eigenimage of the brain, it can be seen that the spongy material (choroid plexus) inside the CSF area has been projected into the eigenimage. This is called a *projection artifact*; a procedure for correcting these kinds of artifacts has been developed by Windham *et al.* [27].

1) Results for Nonlinear Filters

We have calculated ratio images by dividing the second image by the first image (R^{21}), the third by the second (R^{32}), the fourth by the third (R^{43}), the fifth by the fourth (R^{54}), and finally, the first by the fifth (R^{15}). For the simulation and the brain sequence, all images had some pixel intensities very close to zero, so the ratio images had a very large dynamic range, and none of them could be displayed to visualize different features in the scene. This also happened for the R^{43} and R^{54} of the egg phantom. Hence, we have shown R^{15}, the best of the egg ratio images (in terms of segmentation of the desired feature), in Fig. 5(g). The R^{54} for the simulation and the brain are shown in Figs. 2(g) and 8(g). By taking logarithms and arctangents of these ratio images (R^{54} for the simulation and brain, R^{15} for the egg phantom), we generated log-ratio and angle images, which are shown in Figs. 2(h)–(i) 5(h)–(i), and 8(h)–(i).

For reducing the propagated noise and avoiding division by zero, we also considered ratio images formed by dividing each image by the average image. The best ratio, log-ratio, and angle images with respect to the average image are R^{5a}, LR^{5a}, and AG^{5a}, displayed in Figs. 3(g)–(i), 6(g)–(i), and 9(g)–(i), respectively. Note that the propagated noise is reduced, i.e., image SNR and CNR are improved over the above ratio images (see Tables III.b–d), and the dynamic range is not excessive.

For the simulation, the logarithm sufficiently reduced the dynamic range for R^{5a}, but not for R^{54}. The reason is that there are a few pixels in R^{54} with intensities very close to zero; their logarithms tend to minus infinity, resulting in a very large dynamic range for the filtered image. The same problem also occurred in other cases, not shown in this paper for conciseness. This suggests that the logarithm function is not adequate for solving the dynamic range problem. The arctangent function, on the other hand, did sufficiently reduce the dynamic range for all ratio images. There results are in agreement with those found by Wecksung *et al.* [35] for ratio images in the field of remote sensing.

E. Comparison of Theoretical and Experimental Measures

Using the analytical expressions given in Tables I and II, we have *computed* SNR'_ds and CNR'_{du}s of the composite images in each example. Also using the sample mean (2) and variance (3) estimators, we have *experimentally* estimated these quantities. Theoretical and experimental SNR'_ds and CNR'_{du}s are compared in Tables III.b–d. The mathematical predictions and experimental results for the simulated image sequence are in close agreement. This is because the additive noise in simulated images is white with a Gaussian distribution. However, some of the mathematical predictions differ from the experimental results for the egg and brain images. We attribute this to the field and sample (egg and brain) inhomogeneities, which result in inhomogeneous regions for the same material.

Finally, the CPV is investigated by using (8) for eigenimages of the simulation and an agarose phantom; note that this can not be done for the egg phantom and the brain images since the actual partial volumes in the voxels containing multiple tissues are nonconstant and unknown. The partial volume is estimated by

$$\frac{V_{djk}}{V} = \frac{E[TI_{jk}]}{E[T(P^d_{lm})]}. \tag{9}$$

In the simulation, there are three overlapping regions between the desired features (assumed to be the left circular region) and the undesired features (assumed to be the right and top circular regions). For these three regions, estimated partial volumes (EPV) are compared to the original partial volumes (OPV) used in preparing the simulation. For the agarose phantom, three slices at different locations are considered (see Fig. 11), and EPV for six eigenimages, generated by taking one of the pure regions as the desired feature and the other pure region as the interfering feature, is compared to the corresponding OPV which is known from the slice location. Results of both studies are summarized in Tabel IV. It is seen that the eigenimage filter achieves CPV in both cases. Small differences are due to the *estimation* of the signature vectors and mean values, as well as sample and magnetic field inhomogeneities and slice positioning errors which are inevitable in actual MRI studies. This demonstrates the proof given in the Appendix that the eigenimage filter attains CPV.

IV. Discussion of Results

We compared the performance of the eigenimage filter to that of four linear transformations, namely, principal component analysis, matched, modified-matched, and maximum contrast filters, and four nonlinear transformations, namely, target point image method, ratio, log-ratio, and angle image filters. The performance comparison was made by listing mathematical expressions for SNR'_ds and CNR'_{du}s of composite images, investigating CPV by a linear filter, and comparing SNR_d, CNR_{du}, SDF, and CPV on three MRI scene sequences.

A. Linear Filters

We found two difficulties in using PCA for image sequence analysis. First, PCA is size dependent—it enhances large objects better than small objects, since small objects make less contribution to the estimation of the covariance matrix than large objects. Another way of looking at this is that small objects rarely affect GSNR; there is no "motivation" for PCA to enhance them. Therefore, small objects (abnormalities) are not usually visualized in the first PCA image. This suggests that for image analysis and interpretation, all of the PCA images should be used. For instance, although the last PCA image contains a small portion of the total variance, it may visualize features hidden in the original images; using PCA for MR image compression risks losing these hidden features. Second, in general, there is no guarantee that a desired feature will be enhanced and undesired features suppressed in a PCA image. On the other hand, an advantage of PCA is that there is no need to estimate signature vectors. Hence, PCA may be used even though the desired feature (e.g., an abnormality) is not detectable in the original images.

The matched filter maximizes the SNR_d, but since it does not remove any of the interfering features from the scene, it is inappropriate for image segmentation. Its modified version removes any constant (bias) features, at the expense of a decrease in SNR_d. The modified-matched filter is a special case of the eigenimage filter (with the constant vector as the undesired signature vector). Since only changing features are left in a modified-matched image, there is a chance for segmentation of the feature of interest.

In many applications, however, there are nonconstant features which interfere with the observation of the desired object. The eigenimage filter is the optimal linear filter for removing any interfering feature at the lowest possible cost (smallest decrease in SNR_d). As seen from Table I, the SNR_d and CNR'_{du}s in the eigenimage depend on: 1) the SNR of the original images, and 2) the angle between the desired signature vector and its projection onto the undesired subspace. To improve the SNR_d and CNR'_{du}s of the eigenimage, one can improve the quality of the original images [54] and/or optimize MRI protocols and pulse sequence parameters to increase this angle [53]. The important points, however, are that the eigenimage filter always segments the desired feature, and also corrects for partial volume averaging effects.

The maximum contrast filter was originally developed for maximizing CNR between a desired and one undesired feature. There is no unique approach to extend this filter to the case of

TABLE III (a)
EXPERIMENTAL SNR_d, CNR_{du_1}, AND CNR_{du_2} FOR THE ORIGINAL IMAGES

Study	Image Name	SNR_d	CNR_{du_1}	CNR_{du_2}
Simulation	First $T2$	47.993	-16.542	-4.809
	Second $T2$	27.734	-14.389	-19.215
	Third $T2$	16.003	-9.599	-22.409
	Fourth $T2$	10.663	-6.417	-24.535
	$T1$	56.525	1.063	30.383
Egg	First $T2$	37.387	-5.805	15.338
Phantom	Second $T2$	17.507	-10.166	12.044
	Third $T2$	10.959	-12.444	9.446
	Fourth $T2$	5.792	-13.342	3.778
	$T1$	33.999	1.509	-5.139
Brain	First $T2$	21.330	-1.526	-0.751
	Second $T2$	14.030	-1.532	-5.131
	Third $T2$	10.892	-1.208	-8.415
	Fourth $T2$	8.258	-0.909	-10.667
	$T1$	25.893	1.640	12.485

TABLE III (b)
COMPARISON OF MATHEMATICAL AND EXPERIMENTAL SNR_d, CNR_{du_1}, AND CNR_{du_2} FOR LINEAR AND NONLINEAR FILTERS USING A SIMULATED MRI SEQUENCE WITH THREE FEATURES IN THE SCENE

Filter	SNR_d Math.	SNR_d Exp.	CNR_{du_1} Math.	CNR_{du_1} Exp.	CNR_{du_2} Math.	CNR_{du_2} Exp.
PCA_1	-14.190	-14.290	-13.436	-13.511	-48.939	-48.743
Matched	81.470	81.099	-16.631	-16.623	4.093	4.095
Mod-Mat	39.830	39.827	2.597	2.604	40.759	40.477
Eigen	8.128	8.079	8.128	8.064	8.128	8.125
$MMAC_1$	-54.629	-54.709	24.802	24.847	30.678	30.623
$MMAC_2$	6.780	6.834	15.468	15.557	49.190	49.024
$MMAC_3$	-61.897	-61.814	24.517	24.535	24.517	24.456
$MMAC_4$	74.070	73.367	-14.128	-14.107	14.128	14.025
TPI	3.094	3.093	-26.421	-26.607	-54.840	-54.405
ITPI	2.370	2.467	3.097	3.225	3.223	3.356
R^{54}	10.478	10.325	5.343	5.307	12.711	12.541
LR^{54}	17.477	17.582	6.112	6.133	26.057	26.092
AG^{54}	79.630	80.928	6.488	6.506	36.765	36.573
R^{5a}	58.893	59.192	16.020	16.082	43.030	42.018
LR^{5a}	33.907	34.078	16.344	16.397	37.455	36.540
AG^{5a}	145.926	146.673	16.418	16.467	39.967	38.694

multiple interfering features. We considered three possibilities for the case of multiple interfering features: maximized minimum absolute CNR; target point; and inverted target point image methods. The best MMAC image may achieve higher CNR than the eigenimage, TPI, and ITPI. This is achieved at the expense of leaving interfering features in the scene and not correcting for partial volume averaging effects.

B. Nonlinear Filters

We considered ratio, log-ratio, and angle images. We also discussed the possibility of dividing each image by the average image, and then applying logarithm and arctangent functions. We showed that this choice reduces the propagated noise to the transformed images, since noise is suppressed in the average image.

We noted that in the calculation of PCA, ratio, log-ratio, and angle images, there is no need to define signature vectors. Therefore, these transformations can be classified as *unsupervised* methods. In contrast, signature vectors must be defined for matched, modified matched, eigenimage, maximum contrast, and target point filters. Hence, they can be classified as *supervised* methods.

C. Conclusions

From both mathematical and experimental results (Tables I, III.b–d) it is seen that among all linear filters, the matched filter achieves the maximum value for SNR_d and the Maximum CNR filter (MMAC) gives the maximum value for CNR_{du}. However, they do not necessarily segment the desired feature, or correct for partial volume averaging effects. Among

TABLE III (c)
COMPARISON OF MATHEMATICAL AND EXPERIMENTAL SNR_d, CNR_{du_1}, AND CNR_{du_2} FOR LINEAR AND NONLINEAR FILTERS USING AN EGG PHANTOM MRI SEQUENCE WITH THREE FEATURES IN THE SCENE

Filter	SNR_d Math.	SNR_d Exp.	CNR_{du_1} Math.	CNR_{du_1} Exp.	CNR_{du_2} Math.	CNR_{du_2} Exp.
PCA_1	52.32	34.13	−17.33	−9.61	16.14	10.83
Matched	57.86	41.55	−8.13	−4.53	7.87	5.14
Mod−Mat	32.21	25.18	14.72	12.88	−11.23	−8.65
Eigen	9.16	16.08	5.96	13.94	10.72	17.97
$MMAC_1$	−19.02	−11.99	24.74	18.55	−22.40	−16.15
$MMAC_2$	16.18	11.43	−19.68	−18.37	28.16	22.21
$MMAC_3$	−3.04	−4.67	6.11	9.64	11.00	15.45
$MMAC_4$	12.99	10.75	−15.12	−17.38	27.20	24.25
TPI	3.09	2.15	−26.35	−19.46	−30.34	−27.51
ITPI	2.37	2.19	3.10	2.94	3.13	2.85
R^{15}	23.01	28.39	−5.64	−7.68	16.59	21.54
LR^{15}	−8.83	−11.00	−5.70	−8.73	15.51	26.06
AG^{15}	29.58	36.75	−5.80	−8.70	17.33	24.43
R^{5a}	38.80	30.03	13.19	15.18	−19.12	−24.52
LR^{5a}	28.44	21.62	14.20	14.40	−21.11	−20.62
AG^{5a}	111.61	83.88	14.40	13.97	−21.36	−17.67

TABLE III (d)
COMPARISON OF MATHEMATICAL AND EXPERIMENTAL SNR_d, CNR_{du_1}, AND CNR_{du_2} FOR LINEAR AND NONLINEAR FILTERS USING A BRAIN MRI SEQUENCE WITH THREE FEATURES IN THE SCENE

Filter	SNR_d Math.	SNR_d Exp.	CNR_{du_1} Math.	CNR_{du_1} Exp.	CNR_{du_2} Math.	CNR_{du_2} Exp.
PCA_1	39.48	38.48	−0.96	−0.35	1.87	1.82
Matched	39.48	38.43	−0.97	−0.36	1.76	1.73
Mod−Mat	15.53	25.78	1.78	2.60	20.85	28.98
Eigen	3.44	5.13	2.51	4.61	4.03	5.08
$MMAC_1$	−8.44	−9.40	4.54	5.36	21.86	23.05
$MMAC_2$	2.71	4.49	3.87	4.22	25.64	30.76
$MMAC_3$	−19.67	−25.05	3.07	5.06	4.94	5.25
$MMAC_4$	20.75	29.41	−1.90	−2.26	3.05	3.95
TPI	3.09	3.00	−2.81	−3.80	−27.40	−31.67
ITPI	2.37	2.47	1.92	2.86	3.11	3.26
R^{54}	7.59	10.00	1.77	4.15	10.42	11.99
LR^{54}	9.30	13.28	1.87	3.24	17.93	20.87
AG^{54}	36.06	53.05	1.89	2.61	22.94	26.43
R^{5a}	29.18	33.82	4.02	4.55	24.05	25.03
LR^{5a}	11.99	14.64	4.00	3.54	19.04	20.47
AG^{5a}	62.42	74.02	3.99	3.48	21.86	23.44

nonlinear filters, the angle image calculated with respect to the average image provided the maximum value for SNR_d, and the target point image achieved the maximum absolute value for CNR_{du}. The target point image always segments the desired feature; however, the desired feature is actually removed from the scene (it appears black). The inverted target point image overcomes this difficulty, but suffers from low SNR_d and CNR_{du}. The angle image may or may not segment the desired feature; none of these generally corrects for partial volume averaging effects.

Of the transformations discussed in this paper, the *eigenimage filter* is the optimum linear filter which segments the desired feature, removes interfering features, and corrects for partial volume averaging effects. For situations in which SNR_d and CNR'_{du}s of the eigenimages are unsatisfactory, one can improve these quantities by improving the quality of the original images [54] and/or optimizing MRI pulse sequence and parameters [53].

V. APPENDIX

A. Details of Linear Filters

1) Principal Component Analysis (PCA): The weighting vectors for PCA are the normalized eigenvectors of the $n \times n$

TABLE IV
CORRECTION OF PARTIAL VOLUME AVERAGING EFFECTS BY EIGENIMAGE FILTER FOR MRI SEQUENCES OF A SIMULATION AND AN AGAROSE PHANTOM WITH OVERLAPPING FEATURES IN THE SCENE

	Simulation			Agarose Phantom					
EPV	0.4993	0.4775	0.3531	0.5260	0.4847	0.6881	0.3384	0.3715	0.6043
OPV	0.5000	0.5000	0.3333	0.5250	0.4750	0.6750	0.3250	0.3750	0.6250

Estimated Partial Volumes (EPV) are obtained using the sample mean estimator for the overlapping regions of the simulation and the agarose phantom.

Original Partial Volumes (OPV) are values which have been used in preparing the simulation and the agarose phantom, respectively.

sample covariance matrix K, estimated as

$$K_{il} = \frac{1}{M} \sum_{j,k \in \mathrm{ROI}} (P_{jki} - \bar{P}_i)(P_{jkl} - \bar{P}_l), \quad i,l = 1, \cdots, n. \tag{10}$$

Here P_{jki} is the intensity of (j,k)th pixel in the ith image, $\bar{P}_i$ denotes the ith image average gray level in the region of interest (ROI) over which the covariance matrix is estimated, and M is the total number of pixels the ROI. This ROI may be a small region, or the set of all pixels whose intensities are above some specific threshold, or even the whole image. If the ROI is the entire image, the transformation will be driven by overall properties of the tissues within the slice. The order of the PCA images is given by the quantitative distribution of the corresponding eigenvalues of the sample covariance matrix.

2) Matched Filter: The weighting vector for the matched filter is

$$\vec{W} = \vec{d} \tag{11}$$

where $\vec{d}$ is the desired signature vector.

3) Modified-Matched Filter: The weighting vector for the modified matched fiber is

$$\vec{W} = \vec{d} - \left(\frac{1}{n} \sum_{j=1}^{n} d_j \right) \vec{c} \tag{12}$$

where $\vec{c} = [1, 1, \cdots, 1]^T$.

4) Eigenimage Filter: The weighting vector for the eigenimage filter is

$$\vec{W} = \vec{d} - \vec{d}^{\vec{P}} \tag{13}$$

where $\vec{d}^{\vec{P}}$ is the projection of $\vec{d}$ onto the subspace spanned by $\{\vec{S}_k,\ k = 1, \cdots, m, k \neq d\}$ (note that $\vec{S}_d = \vec{d}$). We refer to this subspace as the *undesired* subspace. The weighting vector is computed using a Gram–Schmidt orthogonalization procedure.

5) Maximum Contrast Filter: For a scene with one interfering (undesired) feature, the weighting vector is

$$\vec{W} = \vec{d} - \vec{u}. \tag{14}$$

For a scene with two interfering (undesired) features, the candidate weighting vectors are

1) $\vec{W}_1 = \vec{d} - \vec{u}_1$;
2) $\vec{W}_2 = \vec{d} - \vec{u}_2$;
3) $\vec{W}_3 = \left(\vec{W}_1 \cdot \vec{W}_1 - \vec{W}_1 \cdot \vec{W}_2\right)\vec{W}_2 + (\vec{W}_2 \cdot \vec{W}_2 - \vec{W}_1 \cdot \vec{W}_2)\vec{W}_1$;
4) $\vec{W}_4 = \left(\vec{W}_1 \cdot \vec{W}_1 + \vec{W}_1 \cdot \vec{W}_2\right)\vec{W}_2 - (\vec{W}_2 \cdot \vec{W}_2 + \vec{W}_1 \cdot \vec{W}_2)\vec{W}_1$.

For a scene with more than two interfering (undesired) features, the candidate weighting vectors are given in [55].

B. Details of Nonlinear Filters

1) Target Point Image Method: The target point image (TPI) is defined as

$$\mathrm{TPI}_{jk} = \left\| \vec{P}_{jk} - \vec{d} \right\| = \sqrt{\sum_{i=1}^{n} (P_{jki} - d_i)^2}. \tag{15}$$

Here and in the sequel, subscript $_{jk}$ is used to indicate (j,k)th pixel of the corresponding image. The inverted target point image (ITPI) is defined as

$$\mathrm{ITPI}_{jk} = \frac{1}{\left\| \vec{P}_{jk} - \vec{d} \right\|}. \tag{16}$$

2) Ratio Filter: The ratio image between the lth and mth images $\left(R^{lm}\right)$ is defined as

$$R_{jk}^{lm} = \frac{P_{jkl}}{P_{jkm}} \tag{17}$$

where P_{jkl} and P_{jkm} are respective pixels in the lth and mth images; clearly the mth image must be nonzero everywhere.

3) Log-Ratio Filter: The log-ratio image between the lth and mth images $\left(LR^{lm}\right)$ is defined as

$$LR_{jk}^{lm} = \ln\left[R_{jk}^{lm}\right] = \ln[P_{jkl}] - \ln[P_{jkm}]. \tag{18}$$

4) Angle Image Filter: The angle image between the lth and mth images (AG^{lm}) is defined as

$$AG_{jk}^{lm} = \arctan\left[R_{jk}^{lm}\right]. \tag{19}$$

5) Use of Average Image: It is also possible to generate ratio, log-ratio, and angle images with respect to an average image (A), defined as

$$A_{jk} = \frac{1}{n} \sum_{i=1}^{n} P_{jki}. \tag{20}$$

When using the ratio of the lth image in the sequence to the average image, the resulting images are called ratio over average $\left(R^{la}\right)$, log-ratio over average $\left(LR^{la}\right)$, and angle

over average (AG^{la}), respectively. This choice reduces the propagated noise to the transformed images, because noise is decreased in the average image (see Tables II and III).

C. Rationale for the Application of Ratio Filter to MRI

Considering random distributions for tissue parameters [56], the signal in the ith T2-weighted multiple spin-echo image (S_i) can be represented and simplified as follows:

$$\begin{aligned} S_i &= [N(H)+\sigma_N]\left[1-\exp\left(\frac{-TR}{T1+\sigma_{T1}}\right)\right] \\ &\quad \cdot \exp\left(\frac{-iTE}{T2+\sigma_{T2}}\right) \\ &= [N(H)+\sigma_N]\left[1-\exp\left(\frac{-TR}{T1+\sigma_{T1}}\right)\right] \\ &\quad \cdot \exp\left(\frac{-\frac{iTE}{T2}}{1+\frac{\sigma_{T2}}{T2}}\right) \\ &\simeq [N(H)+\sigma_N]\left[1-\exp\left(\frac{-TR}{T1+\sigma_{T1}}\right)\right] \\ &\quad \cdot \exp\left(-\frac{iTE}{T2}\left(1-\frac{\sigma_{T2}}{T2}\right)\right) \\ &= [N(H)+\sigma_N]\left[1-\exp\left(\frac{-TR}{T1+\sigma_{T1}}\right)\right] \\ &\quad \cdot \exp\left(\frac{iTE}{T2^2}\sigma_{T2}\right)\exp\left(-\frac{iTE}{T2}\right) \\ &= \eta_1\eta_2^i \exp\left(-\frac{iTE}{T2}\right). \end{aligned} \tag{21}$$

Here η_1 and η_2 include the total uncertainty from the tissue parameter distributions. From (21) it is inferred that in a ratio image, most of the signal variations within a specific tissue are compensated. This can improve the SNR of the image (see Tables III.a–d). Proton density contribution to the signal is also suppressed. This may be considered as a disadvantage. However, in a situation where the overall signal is dominated by the proton density contribution (i.e., the proton density contribution dominates the relaxation times contributions to the overall signal), this cancellation can result in visualization of hidden features.

D. Methods of Deriving SNR and CNR Expressions

1) Standard Formula for Noise Propagation: For a random variable z which is a smooth function of uncorrelated random variables $x_1, x_2, \cdots x_m$

$$z = f(x_1, x_2, \cdots, x_m) \tag{22}$$

we have [50], [51]

$$\text{var}(z) = \sigma_z^2 \simeq \sum_{i=1}^{m}\left[\frac{\partial f}{\partial_{x_i}}\Big|_{\overline{x}}\right]^2 \sigma_{x_i}^2 \tag{23}$$

$$\overline{z} \simeq f(\overline{x}_1, \overline{x}_2, \cdots, \overline{x}_m) \tag{24}$$

where $\text{var}(z) =, \sigma_z$, and $\overline{z}$ are the variance, standard deviation, and mean of the random variable z, respectively. Similarly, σ_{x_i}, and $\overline{x}_i$ are the standard deviation and the mean of random variable x_i.

2) Chi-Squared Distribution (SNR Calculation for TPI and ITPI): From the assumed i.i.d. (independent an identically distributed) Gaussian model for MRI noise, it follows that the squared pixel gray levels in the desired (target) ROI of a TPI are scaled (multiplied by σ^2) i.i.d. chi-squared random variables with n degrees of freedom. The chi-squared probability density function is [57]

$$f(x;n) = \begin{cases} \frac{1}{2^{\frac{n}{2}}\Gamma\left(\frac{n}{2}\right)} x^{\frac{n}{2}-1}e^{-\frac{x}{2}} & \text{for } x > 0 \\ 0 & \text{for } x \le 0 \end{cases} \tag{25}$$

where n is the degrees of freedom: $E[x] = n$ and $\text{Var}(x) = 2n$. For a random variable $y = \sigma^2 x$. $E[y] = n\sigma^2$ and $\text{Var}(y) = 2n\sigma^4$.

For TPI we need to calculate the mean and variance of the gray levels inside the DROI of the TPI represented by random variable $z = \sqrt{y} = \sqrt{\sigma^2 x} = \sigma\sqrt{x}$. In [52] we showed that

$$E[z] = \frac{\sigma\sqrt{2}\Gamma\left(\frac{n+1}{2}\right)}{\Gamma\left(\frac{n}{2}\right)} \tag{26}$$

$$\text{Var}(z) = n\sigma^2 - 2\sigma^2\left(\frac{\Gamma\left(\frac{n+1}{2}\right)}{\Gamma\left(\frac{n}{2}\right)}\right)^2. \tag{27}$$

Using (26) and (27), (1) simplifies to

$$\text{SNR}_d = \frac{1}{\sqrt{\frac{n}{2}\left(\frac{\Gamma\left(\frac{n}{2}\right)}{\Gamma\left(\frac{n+1}{2}\right)}\right)^2 - 1}} \tag{28}$$

For ITPI we need to calculate the mean and variance of random variable $w = 1/z$. We showed in [52] that

$$E\left[\frac{1}{z}\right] = \frac{\Gamma\left(\frac{n-1}{2}\right)}{\sigma\sqrt{2}\Gamma\left(\frac{n}{2}\right)} \tag{29}$$

$$\text{Var}\left(\frac{1}{z}\right) = \frac{1}{\sigma^2(n-2)} - \left(\frac{\Gamma\left(\frac{n-1}{2}\right)}{\sigma\sqrt{2}\Gamma\left(\frac{n}{2}\right)}\right)^2. \tag{30}$$

Plugging (29) and (30) in (1) and doing simplifications yield

$$\text{SNR}_d = \frac{1}{\sqrt{\frac{2}{n-2}\left(\frac{\Gamma\left(\frac{n}{2}\right)}{\Gamma\left(\frac{n-1}{2}\right)}\right)^2 - 1}}. \tag{31}$$

E. Correction for Partial Volume Averaging Effects by Eigenimage Filter

We define a vector space whose dimension, n, is equal to the number of images in the sequence. In this vector space, a pixel vector $\vec{P}_{jk} = [P_{jk1}P_{jk2}\cdots P_{jkn}]^T$ is an n-dimensional vector whose elements are the gray levels of the (j,k)th pixels of the images in the sequence. Signature vectors $\{\vec{S}_l = [S_{l1}S_{l2}\cdots S_{ln}]^T, l = 1, \cdots, m\}$ are also n-dimensional vectors whose lth elements specify the desired and undesired features in the lth image, respectively. For a linear transformation, the transformed image (TI) is a weighted

sum (linear combination) of original images in the sequence, i.e.,

$$TI_{jk} = \sum_{i=1}^{n} W_i P_{jki} = \vec{W} \cdot \vec{P}_{jk} \tag{32}$$

where $\vec{W} = [W_1 W_2 \cdots W_n]^T$ is the weighting vector to be determined. Writing (7) in vector notation gives

$$\vec{P}_{jk} = \sum_{l=1}^{m} \left(\frac{V_{ljk}}{V}\right) \vec{S}_l - \vec{\eta}_{jk}. \tag{33}$$

Therefore, we have

$$TI_{jk} = \vec{W} \cdot \vec{P}_{jk} = \sum_{l=1}^{m} \left(\frac{V_{ljk}}{V}\right) \vec{W} \cdot \vec{S}_l + \vec{W} \cdot \vec{\eta}_{jk} \tag{34}$$

and taking expectations (assuming *a priori* known signature vectors) yields

$$E[TI_{jk}] = \sum_{l=1}^{m} \left(\frac{V_{ljk}}{V}\right) \vec{W} \cdot \vec{S}_l + \vec{W} \cdot E[\vec{\eta}_{jk}]. \tag{35}$$

From (8), correction for partial volume averaging effects (CPV) requires

$$E[TI_{jk}] = \left(\frac{V_{djk}}{V}\right) E[\mathcal{T}(P^d_{lm})] = \left(\frac{V_{djk}}{V}\right) \vec{W} \cdot \vec{S}_d, \tag{36}$$

where second equality follows from $E[\vec{W} \cdot \vec{\eta}_{jk}] = \vec{W} \cdot E[\vec{\eta}_{jk}] = 0$, and

$$E[\mathcal{T}(P^d_{lm})] = E\left[\vec{W} \cdot \vec{P}^d_{lm}\right] = \vec{W} \cdot E\left[\vec{P}^d_{lm}\right] = \vec{W} \cdot \vec{S}_d \tag{37}$$

where $\vec{P}^d_{lm}$ represents a pixel vector in the DROI. Subtracting (36) from (35) we obtain

$$\sum_{\substack{l=1 \\ l \neq d}}^{m} \left(\frac{V_{ljk}}{V}\right) \vec{W} \cdot \vec{S}_l = 0 \tag{38}$$

Since (38) must hold for *arbitrary* values of V_{ljk}, we can prove the following.

Claim: CPV is achieved *if and only if*

$$\vec{W} \cdot \vec{S}_l = 0, \quad l = 1, \cdots, m, \; l \neq d. \tag{39}$$

Proof:

If: If (39) holds, then (38) holds, and subtracting this from (35) shows that (36) holds and CPV is attained.

Only If: Let $V_{qjk} = V$ and $V_{ljk} = 0$ for all l such that $l \neq q$ and $l \neq d$. Then (38) requires $\vec{W} \cdot \vec{S}_q = 0$. This argument can be made all for integers$q$ ($1 \leq q \leq m$ and $q \neq d$), hence (39) holds.

Consequently, only the eigenimage filter among all linear filters achieves CPV. Moreover, as we have shown in [58], the filter is optimal in the sense that it achieves the maximum possible SNR_d while correcting for partial volume averaging effects.

VI. Nomenclature

List of Abbreviations

MRI:	Magnetic Resonance Imaging
ROI:	Region of Interest
DROI:	Desired Feature ROI
UROI:	Undesired Feature ROI
SNR:	Signal-to-Noise Ratio
GSNR:	Global SNR
CNR:	Contrast-to-Noise Ratio
SDF:	Segmentation of the Desired Featur
CPV:	Correction for Partial Volume Averaging Effects
EPV:	Estimated Partial Volume
OPV:	Original Partial Volume
TI:	Transformed (composite) Image
PCA:	Principal Component Analysis
Mod-Mat:	Modified Matched
MMAC:	Maximized Minimum Absolute CNR
TPI:	Target Point Image
ITPI:	Inverted Target Point Image
CSF:	Cerebrospinal Fluid

List of Notations

n:	The number of images in the original MRI scene sequence
m:	The number of features in the scene
P_{jk}:	The gray level of the (j,k)th pixel in an original image
P^d_{jk}:	The gray level of the (j,k)th pixel in the DROI
M:	The number of pixels in the DROI
P^u_{jk}:	The gray level of the (j,k)th pixel in the UROI
η_{jk}:	The zero-mean white noise in the image
ρ:	The standard deviation of white noise
$\mathcal{T}(\cdot)$:	The transformation applied to an MRI scene sequence
TI_{jk}:	The gray level of the (j,k)th pixel in a transformed image
PCA_i:	The ith composite image generated by tyhe PCA transformation
MMAC_i:	The ith composite image generated by tyhe MMAC transformation
TPI_{jk}:	The gray level of the (j,k)th pixel in a TPI
ITPI_{jk}:	The gray level of the (j,k)th pixel in an ITPI
R^{lm}:	The ratio image generated by dividing the lth image by the mth image
R^{lm}_{jk}:	The gray level of the (j,k)th in a R^{lm}
R^{la}:	The ratio image generated by dividing the lth image by the average of all original images
R^{ls}_{jk}:	The gray level of the (j,k)th pixel in a R^{la}
LR^{lm}:	The log-ratio image generated by taking the natural logarithm of R^{lm}
LR^{lm}_{jk}:	The gray level of the (j,k)th pixel in a LR^{lm}
LR^{la}:	The log-ratio image generated by taking the nat-

ural logarithm of R^{la}
R^{la}_{jk}: The gray level of the (j,k)th pixel in an LR^{la}
AG^{lm}: The angle image generated by taking the arctangent of R^{lm}
AG^{lm}_{jk}: The gray level of the (j,k)th pixel in an AG^{lm}
AG^{la}: The angle image generated by taking the arctangent of R^{la}
AG^{la}_{jk}: The gray level of the (j,k)th pixel in an LR^{la}
$E[\cdot]$: The expected value operator
$\hat{E}[\cdot]$: An expected value estimator (sample mean)
$Var(\cdot)$: The variance operator
$\hat{Var}(\cdot)$: A variance estimator (sample variance)
SNR_d SNR of the desired feature
CNR_{du}: CNR between the desired and undesired features
$\vec{P}_{jk}$: An n-dimensional vector (pixel vector) whose elements are the gray levels of the (j,k)th pixels of the original images
$\vec{W}$: The weighting vector for a linear transformation (filter)
$\vec{S}_l$: The signature vector for the lth feature
$\vec{S}_d = \vec{d}$: The desired feature signature vector
$\vec{u}_i$: The ith undesired feature signature vector
V_i: The volume of the ith material in a voxel
S_i: The MRI signal from the ith material
V_{ljk}: The partial volume of the lth material in the (j,k)th voxel
V_{djk}: The partial volume of the desired feature in the (j,k)th voxel
V: The total volume of a voxel

REFERENCES

[1] C. H. Hermon and D. Mace, "Use of Karhunen–Loeve transformation in seismic data processing," *Geophys. Prosp.*, vol. 26, pp. 600–626, 1978.
[2] I. F. Jones, "Applications of the Karhunen–Loeve transform in reflection seismology," Ph.D. dissertation, Univ. Brit. Columbia, 1985.
[3] I. F. Jones and S. Levy, "Signal-to-noise ratio enhancement in multichannel seismic data via the Karhunen–Loeve transform," P *Geophys. Prosp.*, vol. 35, pp. 12–32, 1987.
[4] W. J. Done and R. L. Kirlin, "Coherent noise suppression in seismic data using eigenvalue/eigenvector decomposition," *Proc. ICASSP'88*, pp. 908–911, 1988.
[5] G. B. Marchisio, J. V. Pendrel, and B. W. Mattocks "Applications of full and partial Karhunen–Loeve transformation to geophysical image enhancement," in *58th Ann. Int. Soc. Expl. Geophys., Expanded Abstracts.*, 1988, pp. 1266–1269.
[6] H. C. Andrews and C. L. Patterson, "Singular value decomposition and digital image processing," *IEEE Trans. Acoust., Speech, Signal Processing*. vol. ASSP-24, pp. 26–53, 1976.
[7] T. S. Huang and P. M. Narenda, "Image restoration by singular value decomposition," *Appl. Opt.*, vol. 14, pp. 2213–2216, 1975.
[8] B. R. Hunt and O. Kubler, "Karhunen–Loeve multispectral image restoration—Part I: Theory," *IEEE Trans. Acoust., Speech, Signal Processing* vol. ASSP-32, pp. 592–599, 1984.
[9] N. Ahmed and K. R. Rao, *Orthogonal Transforms for Digital Image Processing*. New York: Spring-Verlag, 1975, pp. 189–224.
[10] K. Hsu and M. E. Womble, "Simultaneous noise filtering and data compression of electrocardiogram (ECG)," *Biomed. Sci. Instrum.*, vol. 17, 1981.
[11] S. K. Jenson and F. A. Waltz, "Principal components analysis and canonical analysis in remote sensing," *Annu. Meet. Proc. Amer. Soc. Photogramm. 45th, Tech. Pap.*, vol. 1, pp. 337–348, 1979.
[12] A. Santisteban and L. Munoz. "Principal components of a multispectral image: Application to a geological problem," *IBM J. Res. Develop.*, vol. 22, no. 5, pp. 444–454, 1978.
[13] D. C. Barber, "Digital computer processing of brain scans using principal components," *Phys. Med. Biol.*, vol. 21, no. 5, pp. 792–803, 1976.
[14] U. Schmiedl, D. A. Ortendahl, A. S. Mark, I. Berry, and L. Kaufman, "The utility of principal component analysis for the image display of brain lesions. A preliminary, comparative study," *Mag. Reg. Med.*, vol. 4, pp. 471–486, 1987.
[15] D. A. Ortendahl, "The application of principal component analysis to multivariate MRI data," in *IEEE/Eighth Annu. Conf. Eng. Med. Biol. Soc.*, 1986, pp. 1065–1068.
[16] H. Grahn, N. M. Szeverenyi, M. W. Roggenbuck, F. Delaglio, and P. Geladi, "Data analysis of multivariate magnetic resonance images I. A principal component analysis approach," *Chemomet. Intell. Lab. Syst* vol. 5, pp. 311–322, 1989.
[17] M. Unser and Murray Eden, "Weighted averaging of a set of noisy images for maximum signal-to-noise ratio," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, pp. 890–895, 1990.
[18] D. O. North, "Analysis of the factors which determine signal noise discrimination in radar," RCA Tech. Rep., vol. PTR, pp. 6c, 1943; reprinted in *Proc. IRE*, vol. 51, pp. 1016–1028, 1963.
[19] H. L. Van Trees, *Detection, Estimation, and Modulation Theory*. New York: Wiley, 1968.
[20] S. Chaudhuri, S. Chatterjee, N. Katz, M. Nelson, and M. Goldbaum, "Detection of blood vessels in retinal images using two-dimensional matched filters," *IEEE Trans. Med. Imaging*, vol. 8, pp. 263–269, 1989.
[21] R. A. Kruger and P. Y. Liu, "Digital angiography using a matched filter," *IEEE Trans. Med. Imaging*, vol. MI-1, pp. 16–21, 1982.
[22] S. J. Riederer, A. L. Hall, J. K. Maier, N. J. Pelc, and D. R. Enzmann, "The technical characteristics of matched filtering in digital subtraction angiography," *Med. Phys.* vol. 10 no. 2, pp. 209–217, 1983.
[23] J. B. de Castro, T. A. Tasciyan, J. N. Lee, F. Farzaneh, S. J. Riederer, and R. J. Herfkens, "MR subtraction angiography with matched filter," *JCAT.*, vol. 12, no. 2, pp. 355–362, 1988.
[24] J. W. V. Miller, J. P. Windham, and S. C. Kwatra, "Optimal filtering of radiographic image sequences using simultaneous diagonalization," *IEEE Trans. Med. Imaging.*, vol. MI-3, pp. 116–123, 1984.
[25] J. W. V. Miller, "Techniques for processing image sequences," *Electron. Imaging'88, Int. Electron. Imaging Exposition Conf.*, Advanced Printing of Paper Summaries., vol. 1, pp. 245–250, 1988.
[26] J. P. Windham, M. A. Abd-Allah, D. A. Reimann, J. W. Froelich, and A. M. Haggar, "Eigenimage Filtering in MR Imaging," *J. Comput. Assist. Tomogr.* vol. 12, no. 1, pp. 1–9, 1988.
[27] J. P. Windham, D. J. Peck, D. O. Hearshen, and A. M. Haggar, "Reduction of artifacts in eigenimage filtering of MR images," submitted to *IEEE Trans. Med. Imaging*, 1990.
[28] H. Soltanian-Zadeh, J. P. Windham, and J. M. Jenkins, "Error propagation in eigenimage filtering," *IEEE Trans. Med. Imaging*, vol. MI-4, pp. 405–420, 1990.
[29] J. N. Lee and S. J. Riederer, "The contrast-to-noise in relaxation time, synthetic, and weighted-sum MR images," *Magnet. Reson. Med.* vol. 5, pp. 13–22, 1987.
[30] D. G. Brown, J. N. Lee, R. A. Blinder, H. Z. Wang, S. J. Riederer, and L. W. Nolte, "CNR enhancement in the presence of multiple interfering processes using linear filters," *Magnet Reson. Med.*, pp. 79–96, 1990.
[31] R. B. Buxton, "Target-point image combination for maximizing multiple tissue contrast," *7th Annual. Meet. Soc. Magnet Reson. Med.*, San Francisco, CA, Work in Progress, pp. 157, 1988.
[32] W. K. Pratt, *Digital Image Processing*. New York: Wiley, 1978, pp. 338–342.
[33] K. R. Castleman, *Digital Image Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1979, pp. 348–351.
[34] A. K. Jain, *Fundamentals of Digital Image Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989, pp. 260–261.
[35] G. W. Wecksung and J. R. Breedlove, "Some techniques for digital processing, display and interpretation of ratio images in multispectral remote sensing," *SPIE*, vol. 119, pp. 47–54, 1977.
[36] C. M. Mills, L. E. Crooks, L. Kaufman, and M. Brant-Zawadzki, "Cerebral abnormalities: Use of calculated T1 and T2 magnetic resonance images for diagnosis," *Radiology*, vol. 150, pp. 87–94, 1984.
[37] S. J. Riederer, S. A. Suddarth, S. A. Bobman, J. N. Lee, H. Z. Wang, and J. R. MacFall, "Automatic MR image synthesis: Feasibility studies," *Radiology*, vol. 153, pp. 203–206, 1984.
[38] S. A. Bobman *et al.*, "Cerebral magnetic resonance image synthesis," *AJNR*, vol. 6, pp. 265–269, 1985.
[39] S. A. Bobman, S. J. Riederer, J. N. Lee, S. A. Suddarth, H. Z. Wang, and J. R. MacFall, "Synthesized MR images: Comparison with acquired images," *Radiology*, vol. 155, pp. 731–738, 1985.
[40] W. Kucharczyk *et al.*, "MR technology: Effect on even-echo rephrasing on calculated T2 values and T2 images," *Radiology*, vol. 157, pp. 95–101, 1985.
[41] S. A. Bobman, S. J. Riederer, J. N. Lee, T. Tasciyan, F. Farzaneh, and H. Z. Wang, "Pulse sequence extrapolation with MR image synthesis,"

Radiology, vol. 159, pp. 253–258, 1986.

[42] J. N. Lee, S. J. Riederer, S. A. Bobman, J. P. Johnson, and F. Farzaneh, "The precision of TR extrapolation in magnetic resonance image synthesis," *Med. Phys.*, vol. 3, no. 2, pp. 170–176, 1986.

[43] R. Grauman, H. Fischer, and A. Oppelt, "A new pulse sequence for determining T1 and T2 simultaneously," *Med. Phys.*, vol. 13, no. 5, pp. 644–647, 1986.

[44] S. J. Riederer, S. A. Bobman, J. N. Lee, F. Farzaneh, and H. Z. Wang, "Improved precision in calculated T1 MR images using multiple spin-echo acquisition," *JCAT*, vol. 10, no. 1, pp. 103–110, 1986.

[45] M. E. Masterson, R. Mc Gary, K. Schmitt, and J. A. Koutcher, "Accuracy and reproducibility of image derived relaxation times on a clinical 1.5 T magnetic resonance scanner," *Med. Phys.*, vol. 16, no. 2, pp. 225–233, 1989.

[46] J. R. MacFall, S. J. Riederer, and H. Z. Wang, "An analysis of noise propagation in computed T2, pseudodensity, and synthetic spin-echo images," *Med. Phys.* vol. 13, no. 3, pp. 285–292, 1986.

[47] E. E. McVeigh, R. M. Henkelman, and M. J. Bronskill, "Noise and filtration in magnetic resonance imaging," *Med. Phys.*, vol. 12, no. 5, pp. 586–591, 1985.

[48] F. W. Wehrli, "Signal-to-noise and contrast in MR imaging," in *NMR in Medicine, the Instrumentation and Clinical Applications*, S. R. Thomas and R. L. Dixon, Eds. vol. Monograph no. 14, 1986, pp. 216–228.

[49] H. Stark and J. W. Woods, *Probability, Random Processes, and Estimation Theory for Engineers*. Englewood Cliffs, NJ: Prentice-Hall, 1986.

[50] S. L. Meyer, *Data Analysis for Scientists and Engineers*. New York: Wiley, 1975, pp. 39–48.

[51] J. L. Jaech, *Statistical Analysis of Measurement Error*. New York: Wiley, 1985.

[52] H. Soltanian-Zedah, J. P. Windham, and A. E. Yagle, "Analytical expressions for signal-to-noise ratios for several transformations applied to magnetic resonance image scene sequences," Tech. Rep., Henry Ford Hospital, Detroit, MI, 1991.

[53] H. Soltanian-Zadeh, J. P. Windham, and R. Saigal, "Optimization of MRI protocols and pulse sequences for eigenimage filtering," *in preparation*.

[54] H. Soltanian-Zadeh, J. P. Windham, and D. O. Hearshen, "Pre-processing of MR Image sequence using a new edge-preserving multi-dimensional filter," presented at 10th Annu. Meet. Soc. Mang. Reson. Med. (SMRM), San Francisco, CA, Aug. 1991.

[55] H. Soltanian-Zadeh and J. P. Windham, "Novel and general approach to linear filter design for CNR enhancement of MR images with multiple interfering features in the scene," *J. Electron. Imaging*, vol. 2, pp. 171–182, Apr. 1992.

[56] E. R. McVeigh, M. J. Bronskill, and R. M. Henkelman "Optimization of MR protocols: a statistical decision analysis approach," *Magnet Reson. Med.*, vol. 6, pp. 314–333, 1988.

[57] J. L. Devore, *Probability and Statistics for Engineering and the Sciences* Monterey, CA: Brooks/Cole Publishing Company, 1987, 2nd ed.

[58] H. Soltanian-Zadeh and J. P. Windham, "Optimal linear filter for segmentation of a desired feature from multiple interfering processes," Tech. Rep., Henry Ford Hospital, May 1991.

APPENDIX Q1

H. Soltanian-Zadeh, J.P. Windham and A.E. Yagle, "Optimal Transformation for Correcting Partial Volume Averaging Effects in Magnetic Resonance Imaging," IEEE Trans. Nuclear Science 40(4), 1204-1212, August 1993.

Again, its title describes this paper very well.

Optimal Transformation for Correcting Partial Volume Averaging Effects in Magnetic Resonance Imaging

Hamid Soltanian-Zadeh[1,2], *Member, IEEE*, Joe P. Windham[2], *Member, IEEE*, and Andrew E. Yagle[1], *Member, IEEE*

[1]The University of Michigan, Ann Arbor, MI 48109, and[2]Henry Ford Hospital, Detroit, MI 48202

Abstract

Segmentation of a feature of interest while correcting for partial volume averaging effects is a major tool for identification of hidden abnormalities, fast and accurate volume calculation, and three-dimensional visualization in the field of magnetic resonance imaging (MRI). We present the optimal transformation for simultaneous segmentation of a desired feature and correction of partial volume averaging effects, while maximizing the signal-to-noise ratio (SNR) of the desired feature. It is proved that correction of partial volume averaging effects requires the removal of the interfering features from the scene. It is also proved that correction of partial volume averaging effects can be achieved merely by a linear transformation. It is finally shown that the optimal transformation matrix is easily obtained using the Gram-Schmidt orthogonalization procedure, which is numerically stable. Applications of the technique to MRI simulation, phantom, and brain images are shown. We show that in all cases the desired feature is segmented from the interfering features and partial volume information is visualized in the resulting transformed images.

I. INTRODUCTION

Segmentation of a feature of interest while correcting for partial volume averaging effects is a major tool for image analysis and interpretation in the field of magnetic resonance imaging (MRI). Its applications include identification of hidden abnormalities [1], fast and accurate volume calculation [2], and three-dimensional visualization [3]. In an MRI sequence consisting of several images of the same anatomical site, the image gray levels corresponding to different tissue types, which are functions of intrinsic tissue parameters as well as pulse sequence parameters [4], [5], change characteristically throughout the image sequence and contain information pertaining to partial volume averaging effects. This makes it possible to generate a set of transformed images in which the partial volumes of each feature are visualized.

We derive the optimal transformation for correcting partial volume averaging effects. Optimality is defined as maximizing the signal-to-noise ratio (SNR) of the desired feature, i.e., the feature whose partial volumes are corrected and visualized. In Section II, we first define notations, correction of partial volume averaging effects, SNR, and the optimal transformation. We then establish the relationship between correction of partial volume averaging effects and removal of the interfering features, where we prove that correction of partial volume averaging effects requires removal of the interfering features. We use this relationship to show that correction of partial volume averaging effects can be achieved merely by a linear transformation. Finally, we show that the optimal transformation matrix can be obtained using the Gram-Schmidt orthogonalization procedure. In Section III, we present advantages of the new approach to the previous one for deriving the eigenimage filter [6]. We also present applications of the technique to simulation, phantom, and brain images. Conclusions are given in Section IV. This paper is an extension of the paper presented at the IEEE Medical Imaging Conference in conjunction with the Nuclear Science Symposium [7].

II. METHODS

A. *Problem Formulation*

1) *Notations:* Let $\mathcal{V}$ and $\mathcal{W}$ be n-dimensional and m-dimensional real vector spaces, respectively. Then points in $\mathcal{V}$ and $\mathcal{W}$ are vectors in $\mathcal{R}^n$ and $\mathcal{R}^m$, respectively. Let n be the number of images in the sequence and m be the number of transformed images which equals the number of features (objects) in the scene. Then a *pixel vector* $\vec{\mathbf{P}}_{jk} = [P_{jk1}\ P_{jk2}\ \cdots\ P_{jkn}]^T$ is an n-dimensional vector whose elements are the gray levels of the (j,k)-th pixels in the images in the sequence. The n-dimensional vectors $\{\vec{s}_l = [s_{l1}\ s_{l2}\ \cdots\ s_{ln}]^T,\ l = 1,\cdots,m\}$ whose i-th element defines a specific feature in the i-th image are called *signature vectors*, which are assumed to be linearly independent. This is a reasonable assumption since MRI gray levels in each image are *distinct non-linear* functions of several tissue parameters including proton density (N(H)), spin-lattice (T1) and spin-spin (T2) relaxation times, flow

0018-9499/93$03.00 © 1993 IEEE

velocity (ν), and chemical shift (δ), which are different for different tissues, as well as several pulse sequence parameters including repetition time (TR), echo time (TE), inversion time (TI), and pulse flip angle (θ), which are different for different images in the sequence, [4], [5], making linear dependency of signature vectors very unlikely as long as $n \geq m$. The vector $\vec{s}_d$ ($d \in \{1, 2, \cdots, m\}$) represents the desired feature, and each of the other signature vectors represents an interfering (undesired[1]) feature. A pixel in a transformed image (TI) is a function of all pixels at the same location in the original images, i.e.,

$$TI_{jkd} = \mathcal{T}_d(\vec{\mathbf{P}}_{jk}) \tag{1}$$

where TI_{jkd} is the gray level of the (j,k)-th pixel in the d-th transformed image and $\mathcal{T}_d$ is the function to be found. For a linear transformation, $\mathcal{T}_d(\vec{\mathbf{P}}_{jk}) = \vec{t}_d \cdot \vec{\mathbf{P}}_{jk}$, where $\vec{t}_d = [t_{d1}\ t_{d2}\ \cdots\ t_{dn}]^T$ is the d-th transformation (weighting) vector to be determined.

2) *Correction for Partial Volume Averaging Effects:* The MR signal S from a voxel containing m different materials is given by [8]

$$S = \sum_{l=1}^{m} (\frac{V_l}{V}) S_l \tag{2}$$

where V_l is the volume of the l-th material within the voxel, V is the total volume of the voxel, and S_l is the signal from the l-th material. The gray level P_{jk} of the (j,k)-th pixel (corresponding to the (j,k)-th voxel) in an MR image is given by

$$P_{jk} = E[P_{jk}] + \eta_{jk} = \sum_{l=1}^{m} (\frac{V_{ljk}}{V}) S_l + \eta_{jk} \tag{3}$$

where V_{ljk} is the partial volume of the l-th material in the (j,k)-th voxel, and η_{jk} represents statistical noise which is assumed to be an additive zero-mean white Gaussian noise field, uncorrelated between different scenes of the same MRI sequence, with standard deviation σ (This assumption was previously made and justified in numerous articles including [9]-[14].). Note that $E[P_{jk}]$ is deterministic but unknown, while the noise η_{jk} is stochastic, so that the pixel gray level P_{jk} is the sum of a deterministic value (to be estimated) and noise. We use the notation $E[P_{jk}]$ to denote the original, deterministic value of the pixel gray level, which contains information pertaining to partial volume averaging effects.

Correction of partial volume averaging effects is necessary for robust interpretation and analysis of MR images, as well as for volume calculations. It means that we generate an image whose pixel gray levels, on average, are proportional to the percentages of a specific tissue in the corresponding voxels. Mathematically, this may be translated to generating a transformed image in which

$$E[TI_{jkd}] = (\frac{V_{djk}}{V}) E[\mathcal{T}_d(\vec{\mathbf{P}}^d_{lm})] \tag{4}$$

where $E[TI_{jkd}]$ is the mean value of the (j,k)-th pixel in the transformed image, V_{djk} is the partial volume of the desired material in the (j,k)-th voxel, and $E[\mathcal{T}_d(\vec{\mathbf{P}}^d_{lm})]$ is the mean value of the (l,m)-th pixel in a desired region of interest (ROI) (e.g., the ROI which was used for defining the desired signature vector) from the transformed image. The underlying reason for using the expected value operator in defining correction of partial volume averaging effects by Eq. (4) is to *exclude* the additive noise which contains no information pertaining to these effects. An alternative definition may therefore consist of using a noiseless image model (pixel vector) in Eq. (4) while dropping the expected value operator. Either definition may be used to test correction of partial volume averaging effects. The first definition is usually more appropriate for experimental work, while the second definition is sometimes more appropriate for theoretical development.

3) *Signal-to-Noise Ratio:* Linearly transformed images are linear combinations of the images in the sequence, using different transformation vectors. Since we have m signature vectors each of which can be considered as the desired signature vectors, there are a total of m different transformation vectors resulting in m different transformed images. The pixel gray levels of these linearly transformed images ($\{LTI_d,\ d = 1, \cdots, m\}$) are given by

$$LTI_{jkd} = \sum_{i=1}^{n} T_{id} P_{jki} = \vec{t}_l \cdot \vec{\mathbf{P}}_{jk}, \quad d = 1, \cdots, m \tag{5}$$

where LTI_{jkd} is the gray level of the (j,k)-th pixel in the d-th linearly transformed image, $\vec{t}_d = [T_{1d}\ T_{2d}\ \cdots\ T_{nd}]^T$ is the d-th transformation vector to be determined, and $T = [\vec{t}_1, \vec{t}_2, \cdots, \vec{t}_m]$ is the transformation matrix. For a linear transformation with the transformation vector $\vec{t}_d$, and the presence of an additive zero-mean white noise field with standard deviation σ in the image sequence, the SNR of the desired feature with the signature vector $\vec{s}_d$ is expressed by [15], [16]

$$SNR_d = \frac{\vec{t}_d \cdot \vec{s}_d}{\sigma(\vec{t}_d \cdot \vec{t}_d)^{\frac{1}{2}}}. \tag{6}$$

4) *Optimal Transformation:* We seek a transformation that achieves the following objectives simultaneously:

- Correction of partial volume averaging effects;
- Maximizing SNR of the desired feature.

Theorems 1 and 2 in the next section establish the relationship between correction of partial volume averaging effects, removal of the interfering features, and the linearity of the transformation, and then find the solution.

[1] We use interfering and undesired interchangeably throughout the paper.

B. *Derivation of Solution*

Theorem 1 *(i) For any transformation, correction of partial volume averaging effects requires removal of the interfering features; (ii) correction of partial volume averaging effects can be achieved only by a linear transformation; and (iii) for a linear transformation correction of partial volume averaging effects is equivalent to removal of the interfering features.*

Proof (i) correction of partial volume averaging effects requires that Eq. (4) hold for arbitrary regions of interest in which the expected values (deterministic portion) of the pixel gray levels are convex combinations of the signals from the corresponding overlapping tissues. In the previous section, we described two definitions of correction for partial volume averaging effects. Here, we use the second definition which utilizes the noiseless model for the MRI gray levels (Using the first definition, (ii) cannot be proved in general.). For the noiseless case, a pixel vector can be represented by

$$\vec{\mathbf{P}}_{jk} = \alpha_d \vec{\mathbf{s}}_d + \sum_{\substack{i=1 \\ i \neq d}}^{m} \alpha_i \vec{\mathbf{s}}_i \tag{7}$$

where $\{\alpha_i,\ i = 1, \cdots, m\}$ are partial volumes of the tissues in the corresponding voxel. For this case, Eq. (4) is simplified to

$$TI_{jkd} = \mathcal{T}_d(\vec{\mathbf{P}}_{jk}) = \mathcal{T}_d(\alpha_d \vec{\mathbf{s}}_d + \sum_{\substack{i=1 \\ i \neq d}}^{m} \alpha_i \vec{\mathbf{s}}_i) = \alpha_d \mathcal{T}_d(\vec{\mathbf{s}}_d). \tag{8}$$

Now let $\alpha_i = 0 for\ i \neq l$ and $\alpha_l = 1$ for a fixed $l \neq d$, then Eq. (8) results in

$$\mathcal{T}_d(\vec{\mathbf{s}}_l) = 0. \tag{9}$$

The above argument can me made for each of the interfering features. This shows that correction of partial volume averaging effects requires removal of the interfering features.

(ii) Intuitively, the transformation (demodulator) should be linear since partial volume information is *linearly* modulated into the MRI signal (see Eq. (2)). For a formal proof, consider again the general representation of a noiseless pixel vector given by Eq. (7). Then, Eq. (8) can be written as

$$\begin{aligned} \mathcal{T}_d(\alpha_d \vec{\mathbf{s}}_d + \sum_{\substack{i=1 \\ i \neq d}}^{m} \alpha_i \vec{\mathbf{s}}_i) &= \alpha_d \mathcal{T}_d(\vec{\mathbf{s}}_d) \\ &= \alpha_d \mathcal{T}_d(\vec{\mathbf{s}}_d) + \sum_{\substack{i=1 \\ i \neq d}}^{m} \alpha_i \mathcal{T}_d(\vec{\mathbf{s}}_i). \end{aligned} \tag{10}$$

The last equality uses the necessity of removal of the interfering features proven in (i). Eq. (10) proves the linearity of the desired transformation. Note that although $\{\mathcal{T}_d(\vec{\mathbf{s}}_i) = 0,\ \text{for } i = 1, \cdots, m,\ i \neq d\}$, since $\mathcal{T}_d(\vec{\mathbf{s}}_d) \neq 0$ non-linear terms cannot be added to the right hand side of Eq. (10) except those which are identically zero; the resultant transformation is then effectively linear even though it may have a non-linear appearance.

(iii) Need to show that for a linear transformation removal of the interfering features requires correction of partial volume averaging effects. This is verified by simplifying $E[LTI_{jkd}]$ for an arbitrary pixel vector from the image sequence. Using our assumptions regarding *a priori* known signature vectors, presence of an additive zero-mean white noise field, and considering $\vec{\mathbf{t}}_d$ as the transformation vector for the linear transformation, we have

$$\begin{aligned} E[LTI_{jkd}] &= E[\vec{\mathbf{t}}_d \cdot (\sum_{i=1}^{m} \alpha_i \vec{\mathbf{s}}_i + \vec{\eta}_{jk})] \\ &= E[\sum_{i=1}^{m} \alpha_i \vec{\mathbf{t}}_d \cdot \vec{\mathbf{s}}_i + \vec{\mathbf{t}}_d \cdot \vec{\eta}_{jk}] \\ &= \sum_{i=1}^{m} \alpha_i \vec{\mathbf{t}}_d \cdot \vec{\mathbf{s}}_i \\ &= \alpha_d \vec{\mathbf{t}}_d \cdot \vec{\mathbf{s}}_d \\ &= \alpha_d \mathcal{T}_d(\vec{\mathbf{s}}_d). \end{aligned} \tag{11}$$

In the above we have used the removal of the interfering features and the linearity of the expected value operator. Q.E.D.

Considering the results of Theorem 1, the optimal transformation vector should maximize the SNR_d in Eq. (6) while satisfying the constraints $\vec{\mathbf{t}}_d \cdot \vec{\mathbf{s}}_k = 0$ for $k = 1, \cdots, m,\ k \neq d$. The following Theorem gives the transformation vector and proves its optimality.

Theorem 2 *The solution to the problem*

$$\textbf{Max.} \left[SNR_d = \frac{\vec{\mathbf{t}}_d \cdot \vec{\mathbf{s}}_d}{\sigma(\vec{\mathbf{t}}_d \cdot \vec{\mathbf{t}}_d)^{\frac{1}{2}}} \right] \tag{12}$$

subject to the constraint that

$$\vec{\mathbf{t}}_d \cdot \vec{\mathbf{s}}_k = 0, \quad for \quad k = 1, \cdots, m,\ k \neq d \tag{13}$$

is given by

$$\vec{\mathbf{t}}_d = \vec{\mathbf{s}}_d - \vec{\mathbf{s}}_d^{\,p} \tag{14}$$

where $\vec{\mathbf{s}}_d^{\,p}$ is the projection of $\vec{\mathbf{s}}_d$ onto the subspace spanned by $\{\vec{\mathbf{s}}_k,\ k = 1, \cdots, m,\ k \neq d\}$ (undesired subspace). In addition, $\vec{\mathbf{t}}_d$ can easily be computed using a Gram-Schmidt orthogonalization procedure.

Proof Any n-dimensional vector in general, and $\vec{\mathbf{t}}_d$ in particular, can be represented as [17]

$$\begin{aligned} \vec{\mathbf{t}}_d &= \vec{\mathbf{t}}_d^{\,pfs} + \vec{\mathbf{t}}_d^{\,ofs} \\ &= (\vec{\mathbf{t}}_d^{\,pus} + \vec{\mathbf{t}}_d^{\,ous}) + \vec{\mathbf{t}}_d^{\,ofs} \\ &= \sum_{\substack{i=1 \\ i \neq l}}^{m} c_i \vec{\mathbf{s}}_i + a \vec{\mathbf{s}}_d^{\,o} + \vec{\mathbf{h}} \end{aligned} \tag{15}$$

where $\vec{t}_d^{pf's}$ and $\vec{t}_d^{of's}$ are projections of $\vec{t}_d$ onto the subspace generated by the signature vectors (*feature* subspace) and its orthogonal complement, respectively. Similarly, $\vec{t}_d^{pus}$ and $\vec{t}_d^{ous}$ are projections of $\vec{t}_d^{pf's}$ onto the *undesired* subspace and its orthogonal complement in the feature subspace, respectively. Since $\vec{t}_d^{ous}$ is proportional to $\vec{s}_d^{o}$ (the orthogonal complement, in the *feature* subspace, of the projection of $\vec{s}_d$ onto the undesired subspace), it is replaced by $a\vec{s}_d^{o}$. For convenience, $\vec{t}_d^{pf's}$ is denoted by $\vec{h}$. Finally, a and c_i are scalar coefficients to be determined.

Assuming the undesired signature vectors are linearly independent, the constraint that the undesired signature vectors have zero projections onto the transformation vector requires

$$c_i = 0, \text{ for } i = 1, \cdots, m, \quad i \neq d. \tag{16}$$

This requirement can be mathematically established by substituting Eq. (15) into Eq. (13) followed by an employment of matrix notations.

$$\begin{aligned} \vec{t}_d \cdot \vec{s}_k &= (\sum_{\substack{i=1 \\ i \neq d}}^{m} c_i \vec{s}_i + a\vec{s}_d^{o} + \vec{h}) \cdot \vec{s}_k \\ &= \sum_{\substack{i=1 \\ i \neq d}}^{m} c_i \vec{s}_i \cdot \vec{s}_k = 0, \quad \text{for} \quad k = 1, \cdots, m, \; k \neq d \end{aligned} \tag{17}$$

Define the $n \times (m-1)$ matrix $\mathbf{U}$ by putting the undesired signature vectors $\{\vec{s}_k, \; k = 1, \cdots, m, \; k \neq d\}$ in its columns, and the $(m-1)$-dimensional vector $\vec{C}$ using $\{c_i, i = 1, \cdots, m, \; i \neq d\}$ as its elements, in the same order as the undesired signature vectors used in defining $\mathbf{U}$. Then Eq. (17) can be re-written as

$$(\mathbf{U}^T\mathbf{U})\vec{C} = \vec{0}. \tag{18}$$

Since the undesired signature vectors are assumed to be linearly independent, $\mathbf{U}^T\mathbf{U}$ has rank $(m-1)$ and the unique solution to Eq. (18) is $\vec{C} = \vec{0}$.

Therefore, $\vec{t}_d$ can be written as $\vec{t}_d = a\vec{s}_d^{o} + \vec{h}$. Substituting this representation of $\vec{t}_d$ in Eq. (12), we obtain

$$SNR_d = \frac{\vec{t}_d \cdot \vec{s}_d}{\sigma(\vec{t}_d \cdot \vec{t}_d)^{\frac{1}{2}}} = \frac{a\vec{s}_d^{o} \cdot \vec{s}_d}{\sigma(a^2\|\vec{s}_d^{o}\|^2 + \|\vec{h}\|^2)^{\frac{1}{2}}}. \tag{19}$$

Eq. (19) shows that any non-zero $\vec{h}$ increases the denominator, hence lowers SNR_d. This may indicate that the maximum SNR_d is obtained for $\vec{h} = \vec{0}$. To establish this result in a formal mathematical manner, we note that the necessary condition is

$$\nabla_{\vec{h}} SNR_d = \begin{bmatrix} \frac{-h_1 a\vec{s}_d^{o} \cdot \vec{s}_d}{\sigma(a^2\|\vec{s}_d^{o}\|^2 + \|\vec{h}\|^2)^{\frac{3}{2}}} \\ \vdots \\ \frac{-h_n a\vec{s}_d^{o} \cdot \vec{s}_d}{\sigma(a^2\|\vec{s}_d^{o}\|^2 + \|\vec{h}\|^2)^{\frac{3}{2}}} \end{bmatrix} = \vec{0} \tag{20}$$

which is equivalent to

$$h_i = 0, \text{ for } i = 1, \cdots, n \quad \text{or} \quad \vec{h} = \vec{0} \tag{21}$$

and a sufficient condition is

$$\begin{aligned} \nabla^2_{\vec{h}} SNR_d\Big|_{\vec{h}=\vec{0}} &= \begin{bmatrix} \frac{-a\vec{s}_d^{o} \cdot \vec{s}_d}{\sigma|a|^3\|\vec{s}_d^{o}\|^3} & 0 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & \frac{-a\vec{s}_d^{o} \cdot \vec{s}_d}{\sigma|a|^3\|\vec{s}_d^{o}\|^3} \end{bmatrix} \\ &= \frac{-a\vec{s}_d^{o} \cdot \vec{s}_d}{\sigma|a|^3\|\vec{s}_d^{o}\|^3} \begin{bmatrix} 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} < \mathbf{0} \end{aligned} \tag{22}$$

where " $< \mathbf{0}$ " stands for negative definiteness. This condition is satisfied for any $a > 0$ since

$$\begin{aligned} \vec{s}_d^{o} \cdot \vec{s}_d &= (\vec{s}_d - \vec{s}_d^{p}) \cdot \vec{s}_d \\ &= \|\vec{s}_d\|^2 - \|\vec{s}_d\|^2 \cos^2\theta \\ &= \|\vec{s}_d\|^2 \sin^2\theta > 0 \end{aligned} \tag{23}$$

where θ is the angle between $\vec{s}_d$ and its projection onto the undesired subspace. Therefore,

$$\vec{t}_d = a\vec{s}_d^{o} = a(\vec{s}_d - \vec{s}_d^{p}), \quad a > 0 \tag{24}$$

is a solution. Since a is a scaling factor, it can be set equal to one without loss of generality. The vector $\vec{s}_d^{o}$ can be found by

$$\vec{s}_d^{o} = [\mathbf{I} - \mathbf{U}(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T]\vec{s}_d \tag{25}$$

where $\mathbf{I}$ is the $n \times n$ identity matrix. Eq. (25) yields the solution since it subtracts from $\vec{s}_d$ its projection onto the subspace spanned by the columns of $\mathbf{U}$ (i.e., the undesired subspace). This calculation, however, needs a matrix inversion which may be numerically unstable. Linking the orthogonality of $\vec{s}_d^{o}$ to the undesired subspace (vectors) with that of the Gram-Schmidt orthogonalization [17] (which is numerically stable), it becomes evident that $\vec{s}_d^{o}$ may be found using the Gram-Schmidt method. The key point is that in each step, the Gram-Schmidt orthogonalization removes the projection of the new vector onto the subspace defined by the previous vectors[2]. Therefore, by using $\{\vec{s}_k, \; k = 1, \cdots, m, \; k \neq d\}$ in an arbitrary order and then $\vec{s}_d$ in a Gram-Schmidt orthogonalization, the last output vector will be $(\vec{s}_d - \vec{s}_d^{p})$ which is exactly what we want. The vector $\vec{t}_d$ is always non-zero except for the case in which $\vec{s}_d$ is parallel to the undesired subspace which is very unlikely to happen. Q.E.D.

1) *Existence of the Solution:* In order to guarantee the existence of the transformation vectors, the signature vectors should be linearly independent. This requires that the number of unique images in the sequence (n) be *greater* than or *equal* to the number of signature vectors (m). Here, a unique image is one that is not a linear combination of other images in the sequence.

[2]Details of the Gram-Schmidt orthogonalization may be found in most of the linear algebra textbooks as well as [17].

III. RESULTS

A. *Advantages of Our Approach*

Composite images generated by the optimal transformation are similar to the eigenimages generated by solving generalized eigenvalue problems [6]. It can be mathematically shown that the weighting vector for the eigenimage filtering with one or two interfering features approaches to the transformation vector for our optimal transformation as the regularizing parameter w (defined in [6]) tends to zero. Hence, this work may be considered as a new approach to the derivation of the eigenimage filter as a transformation, with several advantages including:

1. *Well-defined* contrast criteria;
2. No need for the costly *numerical* solutions to the generalized eigenvalue problems;
3. Straightforward *analytical* solution for the general case of multiple interfering features;
4. *Fast* and *numerically stable* calculation of the weighting vectors using the Gram-Schmidt orthogonalization;
5. *Exact* correction for partial volume averaging effects;
6. *Explicit* and *simple* expression for the SNR of the eigenimage, for the case of known or well-estimated signature vectors and additive zero-mean equi-power white noise;
7. Suggesting the normalization of the original images to the standard deviation of noise to yield the *equi-power* white noise case for which the maximum SNR is obtained by the eigenimage filter.

B. *Examples*

We use MRI simulation, phantom, and brain images to illustrate and evaluate the optimal transformation. In generating the simulation, each object (region) is assigned a signature vector equal to that of a normal brain tissue (white matter, gray matter, or cerebrospinal fluid (CSF)). These signature vectors are estimated by averaging image gray levels of normal brain tissues in a four-echo multiple spin-echo protocol with $TE/TR = 25, 50, 75, 100/2500$ msec. As explained in Section II, these signature vectors are non-linear functions of tissue and pulse sequence parameters and thus are linearly independent. Regions of partial volume averaging effects are included into the simulation, where the fractional components of the neighboring objects are known on a pixel-by-pixel basis (They change linearly from 0% to 100% for each tissue, from one row to the next or from one column to the next.). Zero-mean white Gaussian noise with a standard deviation of 0.6 was added to the simulation.

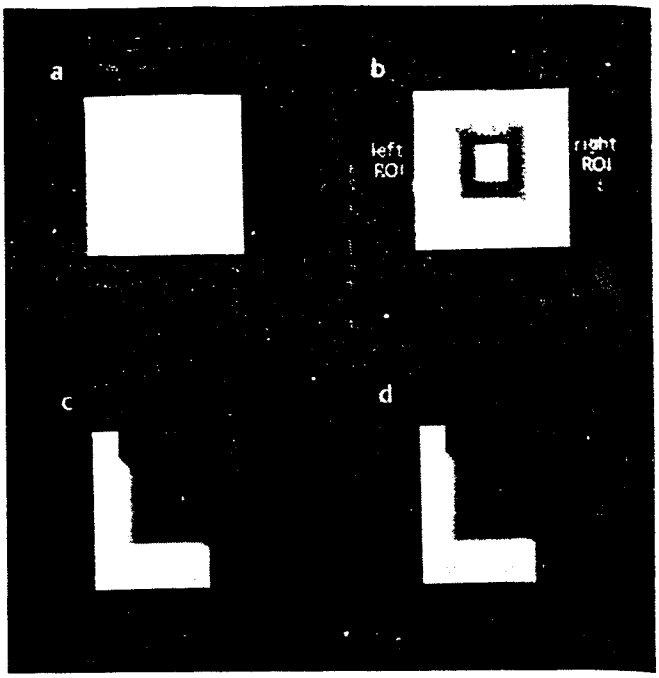


Figure 1: Original images of the simulation. (a)-(d) Four multiple spin-echo images with TE/TR = 25, 50, 75, 100/2500 msec. ROIs used for estimating signature vectors are shown in image (b).

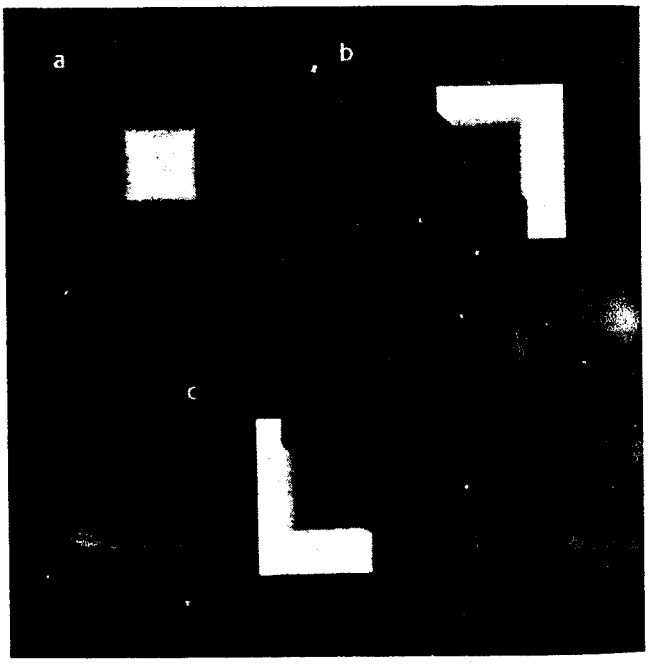


Figure 2: Transformed images of the simulation. (a)-(c) Transformed images for the central region, the region on the right, and the region on the left, respectively.

Table I
Original (org) and Estimated (est) Values of Partial Volumes in the Simulation.

central region	org	10.00%	20.00%	40.00%	55.00%
	est	9.69%	19.81%	41.33%	54.06%
	org	65.00%	80.00%	95.00%	100.00%
	est	65.35%	79.66%	94.84%	100.08%
left region	org	10.00%	20.00%	40.00%	60.00%
	est	10.01%	19.99%	40.23%	59.78%
	org	70.00%	85.00%	95.00%	100.00%
	est	70.05%	84.94%	95.13%	99.93%
right region	org	10.00%	20.00%	30.00%	50.00%
	est	10.46%	20.92%	29.79%	49.65%
	org	60.00%	80.00%	90.00%	100.00%
	est	58.68%	80.30%	89.93%	99.98%

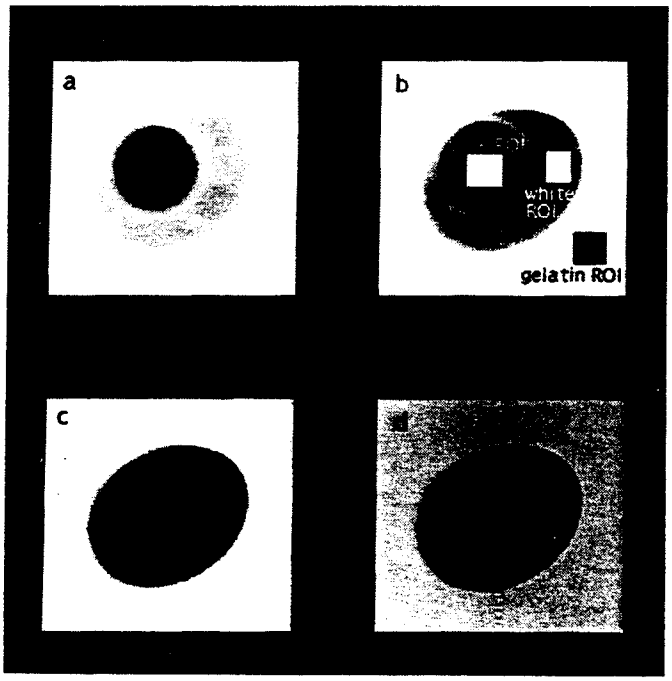


Figure 3: Original images of an egg phantom. (a)-(d) Four multiple spin-echo images with TE/TR = 25, 50, 75, 100/2500 msec. ROIs used for estimating signature vectors are shown in image (b). Note that the zipper artifact in these images is due to the instrument imperfection and thus is not considered as a feature (tissue type). As a result, it is projected to the transformed images shown in Fig. 4.

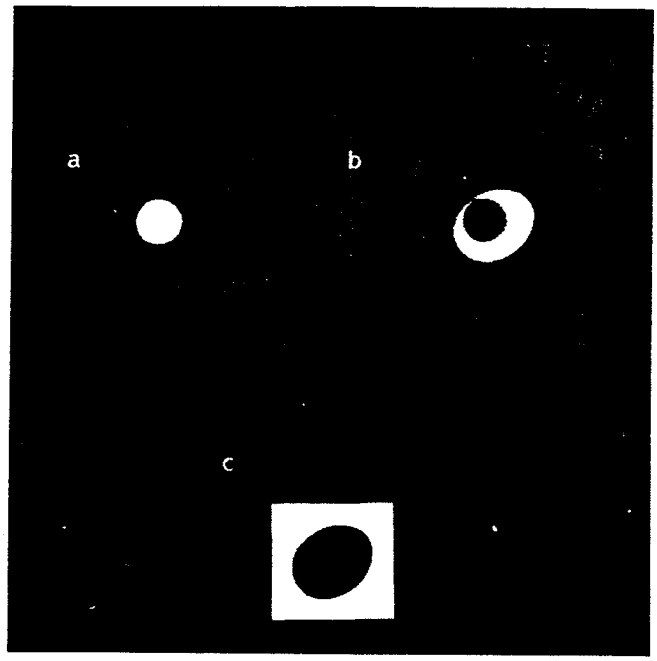


Figure 4: Transformed images of the egg phantom. (a)-(c) Transformed images for egg yolk, egg white, and gelatin, respectively.

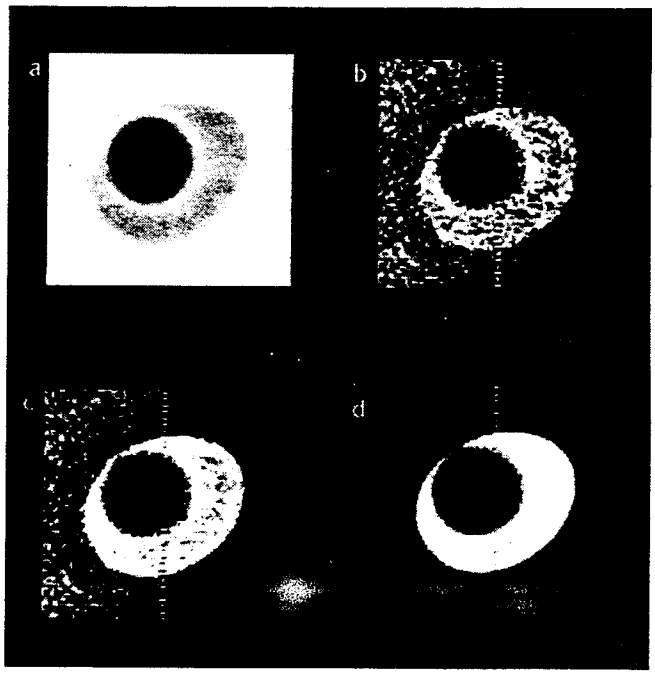


Figure 5: An original and three transformed images for the egg phantom. (a) The first original image of the egg phantom. (b)-(d) Three linearly transformed images for the egg white, all of which correct for partial volume averaging effects but only image (d) maximizes the SNR.

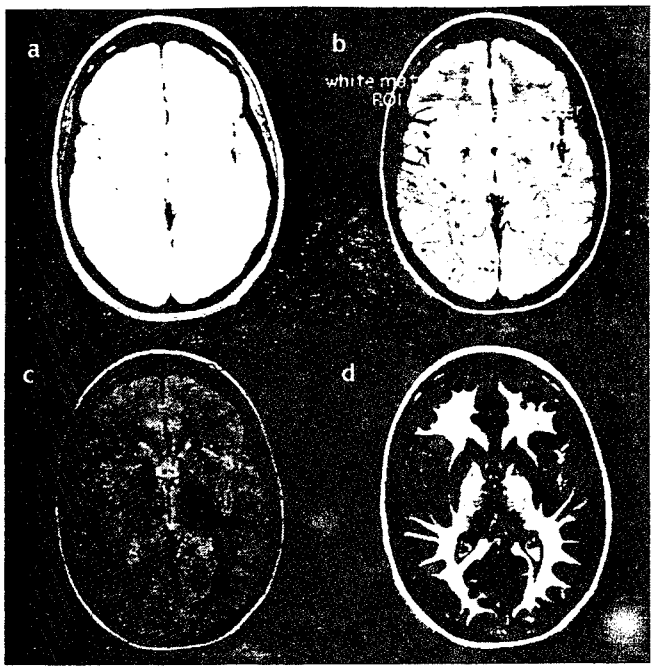


Figure 6: Original images of a human brain. (a)-(c) First three images of a four-echo multiple spin-echo image sequence with TE/TR = 19, 38, 57, 76/1500 msec. (d) An inversion recovery image with TE/TI/TR = 12/519/2000 msec. ROIs used for estimating signature vectors are shown in image (b).

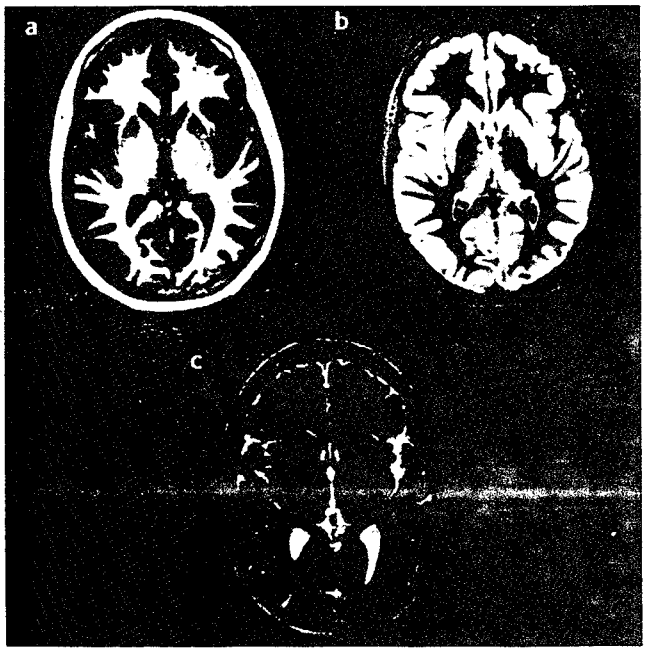


Figure 7: Transformed images of the human brain. (a)-(c) Transformed images for white matter, gray matter, and cerebrospinal fluid (CSF), respectively.

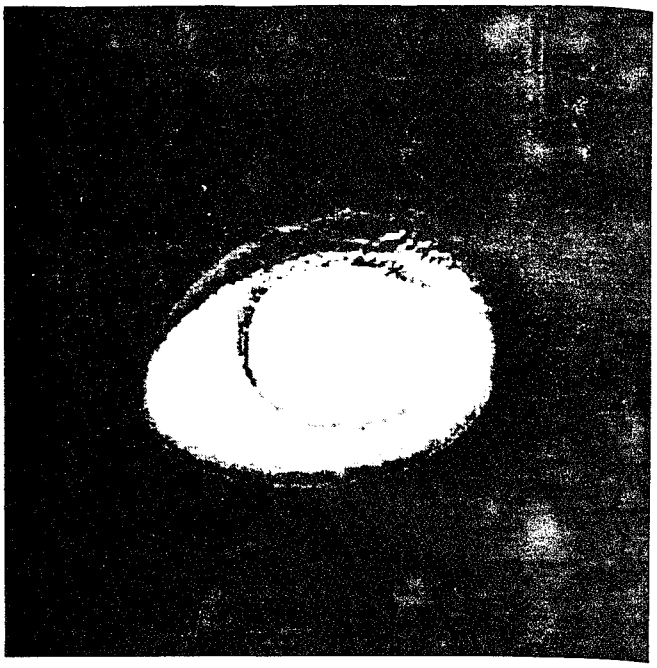


Figure 8: A three-dimensional view of the egg phantom with the partial volume regions visualized.

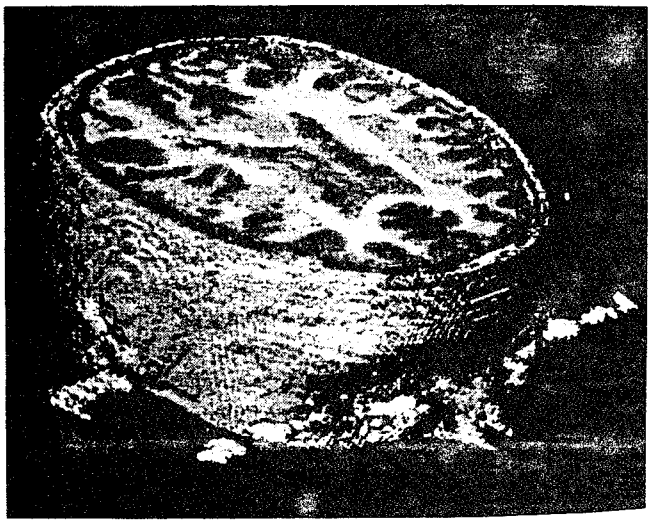


Figure 9: A three-dimensional view of the human brain with the partial volume regions visualized. Note that the non-real objects seen outside of the head are due to the flow artifacts generated during the image acquisition.

The simulated images are shown in Fig. 1. They simulate pure and partial volume regions for white matter, gray matter, and CSF. Regions of interest have been drawn on the pure portions of each object to estimate signature vectors and calculate the transformation matrix. These regions of interest are shown in Fig. 1(b), and the resulting transformed images are shown in Fig. 2. Each transformed image illustrates segmentation of the corresponding desired feature from the interfering features and visualizes its partial volume information. Several partial volume percentages of each feature are estimated using Eq. (4), by drawing horizontal or vertical lines (ROIs) on the partial volume regions and calculating the sample means (estimates of the expected values) for these ROIs and dividing the results by the sample means of ROIs drawn on the corresponding pure regions. The results are summarized in Table I. The original (those used in generating the simulation) and estimated values are in close agreement. Small differences are due to the limitation in the number of noisy pixels available for the estimation.

Original and transformed images of an egg phantom are shown in Figs. 3 and 4, respectively. The partial volumes between egg white and egg yolk, and between egg white and gelatin are not known on a pixel-by-pixel basis, but their averages may be estimated by the water displacement method [2]. These partial volumes are visualized in the transformed images in Fig. 4. An original image of the egg phantom and three linearly transformed images for the egg white are shown in Fig. 5. All of the transformed images satisfy the condition for correcting partial volume averaging effects, i.e., Eq. (4), but they have different SNRs. They illustrate the need for considering SNR maximization in the problem formulation and its effect on the transformed image quality.

Figs. 6 and 7 show the original and transformed images of a normal human brain, respectively. The desired feature is segmented and the partial volumes between different tissues are visualized in the transformed images in Fig. 7.

The transformed images from several slices through the object can be used for three-dimensional (3-D) visualization. An advantage of using these optimally transformed images is that they enable us to segment regions of partial volume averaging effects from regions of pure materials. Figs. 8 and 9 show 3-D images of an egg phantom and a human brain, in which regions of pure and partial volumes are distinguished from each other. Details of the procedure for generating these 3-D images are presented in [18], [19].

IV. SUMMARY AND CONCLUSION

The optimal transformation for simultaneous correction of partial volume averaging effects and maximizing SNR was derived. No linearity assumption was initially made for the transformation. The required properties for the transformation were: (i) correcting for partial volume averaging effects; and (ii) maximizing the SNR of a desired feature in the transformed image. It was shown that property (i) requires removal of the interfering features from the transformed image. Using this, it was proved that property (i) can only be achieved by a linear transformation. It was finally shown that the optimal transformation matrix can be easily and numerically stably obtained using the Gram-Schmidt orthogonalization procedure.

For the mathematical development, we assumed, based on physics of MRI, that the signature vectors were linearly independent. Although, we did not address a situation in which the original signature vectors were linearly dependent, it can be shown that as long as the desired signature vector is linearly independent from the undesired signature vectors, the corresponding optimal transformation vector will correct its partial volumes. However, the partial volume information of the feature whose signature vector is linearly dependent on the rest of the signature vectors cannot be corrected. This is simply because in this situation, there are more unknowns than pieces of information. This result may also be observed from the fact stated at the end of the Proof to Theorem 2, i.e., $\vec{t}_d = \vec{0}$ in this case.

Each column of the optimal transformation matrix is similar to a weighting vector for the eigenimage filter. Therefore, this work may also be considered as a new approach to the derivation of the eigenimage filter, with several advantages including well-defined contrast criteria, straightforward analytical solution along with fast and numerically stable calculation of the weighting vectors, and exact correction for partial volume averaging effects. We believe that the results of the new approach sheds new light on the extended applications of the eigenimage filter to various clinical and industrial problems. It also establishes the applicability of the eigenimage filter to fast and accurate volume determinations.

We used MRI simulation, phantom, and brain images to illustrate and evaluate the optimal transformation. We showed that in all cases the desired feature was segmented from the interfering features and partial volume information was visualized in the resulting transformed images. Using the transformed images, the partial volumes of different features in the simulation were estimated. They were within 1.33% of the actual partial volumes.

In all examples, signature vectors were estimated by averaging several pixels (more than 50) in pure regions of objects (tissues). In a previous publication [20], we showed that this method yields satisfactory estimates of the signature vectors. A stochastic error propagation analysis similar to that of [20] may be used to assess inaccuracy of the transformed images when signature vectors are noisy. A deterministic analysis similar to that of [2] may be used when signature vectors are contaminated, i.e., a pure ROI for each object is not selected. These analyses were beyond the scope of this paper.

The constrained optimization problem formulated in Theorem 2 and its analytical solution can be modified to design other useful transformations. An example may be

found in [21], where we used it to design the optimal linear filter for maximizing the contrast-to-noise ratio (CNR) between a desired feature and multiple interfering features in MRI.

V. ACKNOWLEDGMENT

The authors would like to thank Donald Peck, Steve Ramsey, Lucie Bower, and Linda LeBlanc for their help with programming, data collection, and analysis. They are also grateful to anonymous reviewers whose comments improved presentation of the paper.

VI. REFERENCES

[1] J.P. Windham and D.J. Peck, "Identification of Abnormalities Hidden by Partial Volume Averaging Effects," *Med. Phys.* vol. 17, No. 3, pp. 550, May/June 1990.

[2] D.J. Peck, J.P. Windham, H. Soltanian-Zadeh, and J.R. Roebuck, "A Fast and Accurate Algorithm for Volume Determination in MRI," *Med. Phys.*, vol. 19, No. 3, pp. 599-605, May/June 1992.

[3] J.K. Udupa and G.T. Hermann, *3D Imaging in Medicine.* Ann Arbor: CRC Press Inc., 1991.

[4] F.W. Wehrli, "Principles of Magnetic Resonance," Chapter 1 in *Magnetic Resonance Imaging.* First Edition, Editors: Stark and Bradly, Mosby-Year Book, Inc., St. Louis, MO, pp. 3-23, 1988.

[5] R.E. Hendrick, "Image Contrast and Noise," Chapter 5 in *Magnetic Resonance Imaging*, First Edition, Editors: Stark and Bradly, Mosby-Year Book, Inc., St. Louis, MO, pp. 66-83, 1988.

[6] J.P. Windham, M.A. Abd-Allah, D.A. Reimann, J.W. Froelich, and A.M. Haggar, "Eigenimage Filtering in MR Imaging," *Journal of Computer Assisted Tomography,* vol. 12(1), pp. 1-9, 1988.

[7] H. Soltanian-Zadeh, J.P. Windham, and A.E. Yagle, "Optimal Transformation for Correcting Partial Volume Averaging Effects in Magnetic Resonance Imaging," *Presented at the IEEE Medical Imaging Conference in conjunction with the Nuclear Science Symposium*, Orlando, FL, Oct. 1992.

[8] J.P. Windham, A.M. Haggar, D.O. Hearshen, J.R. Roebuck, and D.A. Reimann, "A Novel Method for Volume Determination Using MR Image Sequence," *Soc. Mag. Res. Imag., Book of Abstracts,* vol. 2, pp. 1081, 1988.

[9] E.R. McVeigh, R.M. Henkelman, and M. J. Bronskill, "Noise and Filtration in Magnetic Resonance Imaging," *Med. Phys.*, vol. 12, No. 5, pp. 586-591, 1985.

[10] F.W. Wehrli, "Signal-to-Noise and Contrast in MR Imaging," *NMR in Medicine, the Instrumentation and Clinical Applications, Editors S. R. Thomas and R. L. Dixon*, volume Monograph No. 14, pp. 216-228, 1986.

[11] J.B. de Castro, T.A. Tasciyan, J.N. Lee, F. Farzaneh, S.J. Riederer, and R.J. Herfkens, "MR Subtraction Angiography with Matched Filter," *JCAT.*, vol. 12(2), pp. 355-362, 1988.

[12] J.N. Lee, and S.J. Riederer, "The Contrast-to-Noise in Relaxation Time, Synthetic, and Weighted-Sum MR Images," *Magn. Reson. Med.*, vol. 5, pp. 13-22, 1987.

[13] D.G. Brown, J.N. Lee, R.A. Blinder, H.Z. Wang, S.J. Riederer, and L.W. Nolte, "CNR Enhancement in the Presence of Multiple Interfering Processes Using Linear Filters," *Magn. Reson. Med.*, vol. 4, pp. 79-96, 1990.

[14] R.B. Buxton and F. Greensite, "Target-Point Combination of MR Images," *Mag. Res. Med.*, vol. 18, No. 1, pp. 102-115, March 1991.

[15] H. Soltanian-Zadeh, "Multi-Dimensional Signal Processing of Magnetic Resonance Scene Sequences," University of Michigan, Ann Arbor, Michigan, PhD Dissertation, 1992.

[16] H. Soltanian-Zadeh, J.P. Windham, D.J. Peck, and A.E. Yagle, "A Comparative Analysis of Several Transformations for Enhancement and Segmentation of Magnetic Resonance Image Scene Sequence." *IEEE Trans. Med. Imag.*, vol. 11, No. 3, pp. 2-19, Sept. 1992.

[17] D.G. Luenberger, *Optimization by Vector Space Methods.* New York: John Wiley & Sons Inc., 1969.

[18] J.P. Windham, D.J. Peck, and H. Soltanian-Zadeh, "Segmentation of Soft Tissue for 3D Visualization with MR Imaging." *Presented at the 78th Scientific Assembly and Annual Meeting of the Radiological Society of North America (RSNA)*, Chicago, IL , Nov. 1992.

[19] J.P. Windham, D.J. Peck, and H. Soltanian-Zadeh, "Segmentation of Soft Tissue for 3D Visualization with MR Imaging." *To be Submitted to Med. Phys.*, 1993.

[20] H. Soltanian-Zadeh, J.P. Windham, and J.M. Jenkins, "Error Propagation in Eigenimage Filtering," *IEEE Trans. Med. Imag.*, vol. 9(4), pp. 405-420, 1990.

[21] H. Soltanian-Zadeh and J.P. Windham, "Novel and General Approach to Linear Filter Design for CNR Enhancement of MR Images with Multiple Interfering Features in the Scene," Journal of Electronic Imaging, vol. 1, No. 2, pp. 171-182, April 1992.

APPENDIX Q2

H. Soltanian-Zadeh, J.P. Windham and A.E. Yagle, "Optimal Transformation for Correcting Partial Volume Averaging Effects in Magnetic Resonance Imaging," IEEE 1992 Medical Imaging Conference, Orlando FL, Oct. 25-31, 1992, pp. 1289-91.

This is the conference paper version of Appendix Q1.

Optimal Transformation for Correcting Partial Volume Averaging Effects in Magnetic Resonance Imaging

Hamid Soltanian-Zadeh[1,2], *Joe P. Windham*[2], *and Andrew E. Yagle*[1]
[1]The University of Michigan, Ann Arbor, MI 48109, and [2]Henry Ford Hospital, Detroit, MI 48202

Abstract

Segmentation of a feature of interest while correcting for partial volume averaging effects is a major tool for identification of hidden abnormalities, fast and accurate volume calculation, and three-dimensional visualization in the field of magnetic resonance imaging (MRI). We present the optimal transformation for simultaneous segmentation of a desired feature (SDF) and correction of partial volume averaging effects (CPV), while maximizing the signal-to-noise ratio (SNR) of the desired feature. It is proved that CPV requires the removal of the interfering features from the scene (RIF). It is also proved that CPV can be achieved merely by a linear transformation. It is finally shown that the optimal transformation matrix is easily obtained using the Gram-Schmidt orthogonalization procedure, which is numerically stable. Applications of the technique to MRI simulation, phantom, and brain images were shown in the presentation. We showed that in all cases the desired feature was segmented from the interfering features and partial volume information was visualized in the resulting transformed images.

I. INTRODUCTION

Segmentation of a feature of interest while correcting for partial volume averaging effects is a major tool for MRI image analysis and interpretation. Its applications include identification of hidden abnormalities [1], fast and accurate volume calculation [2], and three-dimensional visualisation [3]. In an MRI sequence consisting of several images of the same anatomical site, the image gray levels corresponding to different tissue types change characteristically throughout the image sequence and contain information pertaining to partial volume averaging effects. This makes it possible to generate a set of transformed images in which the partial volumes of each feature are visualized.

We derive the optimal transformation for correcting partial volume averaging effects. Optimality is defined as maximizing the SNR of the desired feature, i.e., the feature whose partial volumes are corrected and visualized. In Section II, after defining notations and CPV, we establish the relationship between CPV and RIF. This relationship is used to show that CPV can be achieved merely by a linear transformation. Then we find the optimal transformation matrix using the Gram-Schmidt orthogonalization procedure. In Section III, we present advantages of the new approach to the previous one for deriving the eigenimage filter [4]. In the presentation, we showed applications of the technique to simulation, phantom, and brain images. Conclusions are given in Section IV.

II. METHODS

A. *Problem Formulation*

1) *Notations:* Let $\mathcal{V}$ and $\mathcal{W}$ be n-dimensional and m-dimensional real vector spaces, respectively. Then points in $\mathcal{V}$ and $\mathcal{W}$ are vectors in $\mathcal{R}^n$ and $\mathcal{R}^m$, respectively. Let n be the number of images in the sequence and m be the number of transformed images which equals the number of features (objects) in the scene. Then a *pixel vector* $\vec{\mathbf{P}}_{jk} = [P_{jk1}\ P_{jk2}\ \cdots\ P_{jkn}]^T$ is an n-dimensional vector whose elements are the gray levels of the (j,k)-*th* pixels in the images in the sequence. The n-dimensional vectors $\{\vec{s}_l = [s_{l1}\ s_{l2}\ \cdots\ s_{ln}]^T,\ l = 1,\cdots,m\}$ whose i-*th* element defines a specific feature in the i-*th* image are called *signature vectors*. The vector $\vec{s}_d$ $(1 \le d \le m)$ represents the desired feature, and each of the other signature vectors represents an interfering feature. A pixel in a transformed image (TI) is a function of all pixels at the same location in the original images, i.e.,

$$TI_{jk} = \mathcal{T}(\vec{\mathbf{P}}_{jk}) \tag{1}$$

where TI_{jk} is the gray level of the (j,k)-*th* pixel in the transformed image, and $\mathcal{T}$ is the function to be found. For a linear transformation, $\mathcal{T}(\vec{\mathbf{P}}_{jk}) = \vec{\mathbf{t}} \cdot \vec{\mathbf{P}}_{jk}$, where $\vec{\mathbf{t}} = [t_1\ t_2\ \cdots\ t_n]^T$ is the transformation (weighting) vector to be determined.

2) *Correction for Partial Volume Averaging Effects:* The MR signal S from a voxel containing m different ma-

0-7803-0883-2/93/$3.00 ©1993 IEEE

terials is given by [5]

$$S = \sum_{i=1}^{m} (\frac{V_i}{V}) S_i \tag{2}$$

where V_i is the volume of the i-th material within the voxel, V is the total volume of the voxel, and S_i is the signal from the i-th material. The gray level P_{jk} of the (j,k)-th pixel (corresponding to the (j,k)-th voxel) in an MR image is given by

$$P_{jk} = E[P_{jk}] + \eta_{jk} = \sum_{l=1}^{m} (\frac{V_{ljk}}{V}) S_l + \eta_{jk} \tag{3}$$

where V_{ljk} is the partial volume of the l-th material in the (j,k)-th voxel, and η_{jk} represents statistical noise which is assumed to be an additive zero-mean white Gaussian noise field with standard deviation σ. Note that $E[P_{jk}]$ is deterministic but unknown, while the noise η_{jk} is stochastic, so that the pixel gray level P_{jk} is the sum of a deterministic function (to be estimated) and noise. We use the notation $E[P_{jk}]$ to denote the original, deterministic value of the pixel gray level.

CPV means that we generate an image whose pixel gray levels are proportional to the percentage of a specific tissue in each voxel on average. Mathematically, this translates to generating a transformed image (TI) in which

$$E[TI_{jk}] = (\frac{V_{djk}}{V}) E[T(\vec{\mathbf{P}}^d_{lm})] \tag{4}$$

where $E[TI_{jk}]$ is the mean value of the (j,k)-th pixel in the transformed image, V_{djk} is the partial volume of the desired material in the (j,k)-th voxel, and $E[T(\vec{\mathbf{P}}^d_{lm})]$ is the mean value of the (l,m)-th pixel in the desired region of interest (ROI) of the transformed image. This correction is necessary for robust interpretation and analysis of MR images, as well as for volume calculations.

3) *Signal-to-Noise Ratio:* Linearly transformed images are linear combinations of the images in the sequence, using different transformation vectors. Since we have m signature vectors each of which can be considered as the desired signature vectors, there are a total of m different transformation vectors resulting in m different transformed images. The pixel gray levels of these linearly transformed images ($\{LTI_l,\ l = 1, \cdots,\ m\}$) are given by

$$LTI_{jkl} = \sum_{i=1}^{n} T_{il} P_{jki} = \vec{\mathbf{t}}_l \cdot \vec{\mathbf{P}}_{jk}, \quad l = 1, \cdots,\ m \tag{5}$$

where LTI_{jkl} is the gray level of the (j,k)-*th* pixel in the l-th linearly transformed image, $\vec{\mathbf{t}}_l = [T_{1l}\ T_{2l}\ \cdots\ T_{nl}]^T$ is the l-th transformation vector to be determined, and $T = [\vec{\mathbf{t}}_1, \vec{\mathbf{t}}_2, \cdots, \vec{\mathbf{t}}_m]$. For a linear transformation with the transformation vector $\vec{t}_l$, and the presence of an additive zero-mean white noise field with standard deviation σ in the image sequence, the SNR of the desired feature with the signature vector $\vec{\mathbf{s}}_l$ is expressed by [6]

$$SNR_l = \frac{\vec{\mathbf{t}}_l \cdot \vec{\mathbf{s}}_l}{\sigma(\vec{\mathbf{t}}_l \cdot \vec{\mathbf{t}}_l)^{\frac{1}{2}}}. \tag{6}$$

4) *Optimal Transformation:* We seek a transformation that achieves the following objectives simultaneously:

- CPV, i.e., satisfying Eq. (4);
- Maximizing SNR of the desired feature, i.e., maximizing Eq. (6).

Theorems 1 and 2 in Section B stablish the relationship between CPV, RIF, and the linearity of the transformation, and then find the solution.

B. *Derivation of Solution*

Theorem 1 *(i) For any transformation, CPV requires RIF; (ii) CPV can be achieved only by a linear transformation; and (iii) for a linear transformation CPV is equivalent to RIF.*

Proof Due to the page limitations, we do not give the proof here; it is given in [7].

Considering the results of Theorem 1, the optimal weighting vector should maximize the above SNR_l while satisfying the constraints $\vec{\mathbf{t}}_l \cdot \vec{\mathbf{s}}_k = 0$, for $k = 1, \cdots, m,\ k \neq l$. The following Theorem gives the transformation vector and proves its optimality.

Theorem 2 *The solution to the problem*

$$\textbf{Max.} \left[SNR_l = \frac{\vec{\mathbf{t}}_l \cdot \vec{\mathbf{s}}_l}{\sigma(\vec{\mathbf{t}}_l \cdot \vec{\mathbf{t}}_l)^{\frac{1}{2}}} \right] \tag{7}$$

subject to the constraint that

$$\vec{\mathbf{t}}_l \cdot \vec{\mathbf{s}}_k = 0, \quad for \quad k = 1, \cdots, m,\ k \neq l \tag{8}$$

is given by

$$\vec{\mathbf{t}}_l \ = \ \vec{\mathbf{s}}_l - \vec{\mathbf{s}}_l^{\mathcal{P}} \tag{9}$$

where $\vec{\mathbf{s}}_l^{\mathcal{P}}$ is the projection of $\vec{\mathbf{s}}_l$ onto the subspace spanned by $\{\vec{\mathbf{s}}_k,\ k = 1, \cdots, m,\ k \neq l\}$. In addition, $\vec{\mathbf{t}}_l$ can easily be computed using a Gram-Schmidt orthogonalization procedure.

Proof Due to the page limitations, we do not give the proof here; it is given in [7], [8].

1) *Existence of the Solution:* In order to guarantee the existence of the transformation vectors, the number of unique images in the sequence (n) must be *equal to or greater* than the number of signature vectors (m). Here, a unique image is one that is not a linear combination of other images in the sequence, i.e., it contains information not present in other images.

III. RESULTS

Composite images generated by the optimal transformation are similar to the eigenimages generated by solving generalized eigenvalue problems [4]. Hence, this work may be considered as a new approach to the derivation of the eigenimage filter as a transformation, with several advantages including:

1. *Well-defined* contrast criterion;
2. No need for the costly *numerical* solutions to the generalized eigenvalue problems;
3. Straightforward *analytical* solution for the general case of multiple interfering features;
4. *Fast* and *numerically stable* calculation of the weighting vectors using the Gram-Schmidt orthogonalization;
5. *Exact* correction for partial volume averaging effects;
6. *Explicit* and *simple* expression for the SNR of the eigenimage, for the case of known or well-estimated signature vectors and additive zero-mean equi-power white noise;
7. Suggesting the normalization of the original images to the standard deviation of noise to yield the *equi-power* white noise case for which the maximum SNR is obtained by the eigenimage filter.

A. *Examples*

Applications of the optimal transformation to the MRI simulation, phantom, and brain images were shown in the presentation, we do not include them here due to the page limitations. We showed that in all cases the desired feature was segmented from the interfering features and partial volume information was visualized in the resulting transformed images. Volume calculations for agarose and egg phantoms were also presented.

IV. SUMMARY

The optimal transformation for simultaneous CPV and maximizing SNR was derived. No linearity assumption was initially made for the transformation. The required properties for the transformation were: (*i*) maximizing the SNR of a desired feature in the transformed image; and (*ii*) correcting for partial volume averaging effects. It was shown that property (*ii*) requires removal of the interfering features from the transformed image. Using this, it was shown that property (*ii*) can only be achieved by a linear transformation. It was finally proved that the optimal transformation matrix is easily and numerically stably obtained using the Gram-Schmidt orthogonalization procedure.

Each column of the optimal transformation matrix is similar to a weighting vector for the eigenimage filter. Therefore, it may be considered as a new derivation of the eigenimage filter with several advantages. We believe that the results of the new derivation sheds new light on the extended applications of the eigenimage filter to various clinical or industrial problems. It also establishes the applicability of the eigenimage filter for fast and accurate *in vivo* volume determinations.

V. ACKNOWLEDGMENT

The authors would like to thank Donald Peck, Steve Ramsey, Lucie Bower, and Linda LeBlanc for their help with programming, data collection, and analysis.

VI. REFERENCES

[1] J.P. Windham and D.J. Peck, "Identification of Abnormalities Hidden by Partial Volume Averaging Effects," *Med. Phys.* vol. 17, No. 3, pp. 550, May/June 1990.

[2] D.J. Peck, J.P. Windham, H. Soltanian-Zadeh, and J.R. Roebuck, "A Fast and Accurate Algorithm for Volume Determination in MRI," *Med. Phys.*, vol. 19, No. 3, pp. 599-605, May/June 1992.

[3] J.K. Udupa and G.T. Hermann, *3D Imaging in Medicine.* Ann Arbor: CRC Press Inc., 1991.

[4] J.P. Windham, M.A. Abd-Allah, D.A. Reimann, J.W. Froelich, and A.M. Haggar, "Eigenimage Filtering in MR Imaging," *Journal of Computer Assisted Tomography,* vol. 12(1), pp. 1-9, 1988.

[5] J.P. Windham, A.M. Haggar, D.O. Hearshen, J.R. Roebuck, and D.A. Reimann, "A Novel Method for Volume Determination Using MR Image Sequence," *Soc. Mag. Res. Imag., Book of Abstracts,* vol. 2, pp. 1081, 1988.

[6] H. Soltanian-Zadeh, J.P. Windham, D.J. Peck, and A.E. Yagle, "A Comparative Analysis of Several Transformations for Enhancement and Segmentation of Magnetic Resonance Image Scene Sequence." *IEEE Trans. Med. Imag.*, vol. 11, No. 3, pp. 2-19, Sept. 1992.

[7] H. Soltanian-Zadeh, "Multi-Dimensional Signal Processing of Magnetic Resonance Scene Sequences." PhD Dissertation, University of Michigan, Ann Arbor, Michigan, 1992.

[8] H. Soltanian-Zadeh and J.P. Windham, "Novel and General Approach to Linear Filter Design for CNR Enhancement of MR Images with Multiple Interfering Features in the Scene," Journal of Electronic Imaging, vol. 1, No. 2, pp. 171-182, April 1992.

APPENDIX R1

H. Soltanian-Zadeh, R. Saigal, J.P. Windham, A.E. Yagle, and D.O. Hearshen, "Optimization of MRI Protocols and Pulse Sequence Parameters for Eigenimage Filtering," revision submitted to IEEE Trans. Medical Imaging, July 1993.

This paper proposes a procedure for optimizing the acquisition of MRI scene sequences, if eigenimage filtering (see Appendix P) is then used to process the MRI scene sequence.

Optimization of MRI Protocols and Pulse Sequence Parameters for Eigenimage Filtering

Hamid Soltanian-Zadeh, *Member, IEEE*, Romesh Saigal, Joe P. Windham, *Member, IEEE*, Andrew E. Yagle, *Member, IEEE*, and David O. Hearshen

Abstract—**The eigenimage filter generates a composite image in which a desired feature is segmented from interfering features. The signal-to-noise ratio (SNR) of the eigenimage equals its contrast-to-noise ratio (CNR) and is directly proportional to the dissimilarity between the desired and interfering features. Since image gray levels are analytical functions of magnetic resonance imaging (MRI) parameters, it is possible to maximize this dissimilarity by optimizing these parameters. For optimization, we consider four MRI pulse sequences: multiple spin-echo (MSE); spin-echo (SE); inversion recovery (IR); and gradient-echo (GE). We use the mathematical expressions for MRI signals along with intrinsic tissue parameters to express the objective function (normalized SNR of the eigenimage) in terms of MRI parameters. The objective function along with a set of diagnostic or instrumental constraints define a multidimensional nonlinear constrained optimization problem, which we solve by the fixed point approach. The optimization technique is demonstrated through its application to phantom and brain images. We show that the optimal pulse sequence parameters for a sequence of four MSE and one IR images almost doubles the smallest normalized SNR of the brain eigenimages, as compared to the conventional brain protocol.**

I. Introduction

A. Background and Motivation

THE EIGENIMAGE FILTER maximizes the projection of a desired feature while minimizing the projections of the undesired (interfering) features in a composite image called an eigenimage (EI) [1]. It has been shown that the eigenimage is the unique transformation for magnetic resonance imaging (MRI) scene segmentation that maximizes signal-to-noise ratio (SNR) while correcting for partial volume averaging effects [2], [3]. Since in the eigenimage the desired feature appears bright, the interfering features appear dark, and the partial volumes of the desired feature are visualized, viewing the eigenimage would be helpful for a better interpretation of the MRI scene. Fig. 1 illustrates the application of the eigenimage filter for segmenting gray matter from white matter and cerebrospinal fluid (CSF) in brain images.

Mathematically, the eigenimage is a weighted sum of the images in the sequence. To explain the derivation of the eigenimage filter, we use the n-dimensional Euclidean vector space $\mathcal{R}^n$, where n is the number of images in the MRI scene sequence. For example, when dealing with an MRI scene sequence consisting of a $T1$-weighted and four $T2$-weighted spin-echo images, we use a five-dimensional Euclidean vector space. Then, an MRI scene sequence is represented by *pixel vectors*. A pixel vector $\vec{\mathbf{P}}_{jk} = [P_{jk1}P_{jk2}\cdots P_{jkn}]^T$ is a vector whose elements are the corresponding gray levels of the (j,k)-th pixels in the MR images (see Fig. 1). The image size determines the number of these pixel vectors, e.g., for 256×256 images there are 2^{16} pixel vectors. The MRI characteristics of tissue types are represented by *signature vectors*. For image analysis, one is normally interested in clearly visualizing one of the tissue types (referred to as the desired feature), while other tissue types (referred to as the undesired or interfering features) interfere with its visualization. A *desired signature vector* $\vec{\mathbf{d}} = [d_1 d_2 \cdots d_n]^T$ is defined as a vector whose i-th element is the average gray level of the desired feature in the i -th image. *Undesired (interfering) signature vectors* $\vec{\mathbf{u}}_i = [u_{1i}u_{2i}\cdots u_{ni}]^T$, $1 \le i \le m'$ (m' is the number of interfering features), are similarly defined for the interfering features (see Fig. 1).

A pixel in the eigenimage (EI_{jk}) is a linear combination of all pixels at the same location in the MR images, i.e.,

$$\mathrm{EI}_{jk} = \sum_{i=1}^{n} e_i P_{jki} = \vec{\mathbf{e}} \cdot \vec{\mathbf{P}}_{jk} \tag{1}$$

where EI_{jk} is the gray level of the (jk)-th pixel in the eigenimage, and $\vec{\mathbf{e}} = [e_1 e_2 \cdots e_n]^T$ is the weighting vector to be determined.

To determine the weighting vector, the SNR of the desired feature is maximized

$$\mathbf{Max.}\left[\mathrm{SNR} = \frac{\vec{\mathbf{e}} \cdot \vec{\mathbf{d}}}{\sigma(\vec{\mathbf{e}} \cdot \vec{\mathbf{e}})^{\frac{1}{2}}}\right] \tag{2}$$

subject to the constraint that

$$\vec{\mathbf{e}} \cdot \vec{\mathbf{u}}_i = 0, \text{ for } \quad i = 1, \ldots, m'. \tag{3}$$

Manuscript received January 19, 1993; revised August 25, 1993. This work was supported in part by NIH grant 1R29-CA61263. The associate editor responsible for coordinating the review of this paper and recommending its publication was E. M. Haacke.

H. M. Soltanian-Zadeh is with the Department of Electrical Engineering and Computer Science, the University of Michigan, Ann Arbor, MI 48109 and the Department of Diagnostic Radiology and Medical Imaging, Henry Ford Hospital, Detroit, MI 48202.

R. Saigal is with the Department of Industrial and Operation Engineering, the University of Michigan, Ann Arbor, MI 48109.

J. P. Windham is with the Department of Diagnostic Radiology and Medical Imaging, Henry Ford Hospital, Detroit, MI 48202.

A. E. Yagle is with the Department of Electrical Engineering and Computer Science, the University of Michigan, Ann Arbor, MI 48109.

D. O. Hearshen is with the Department of Diagnostic Radiology and Medical Imaging, Henry Ford Hospital, Detroit, Michigan 48202.

IEEE Log Number 9215306D.

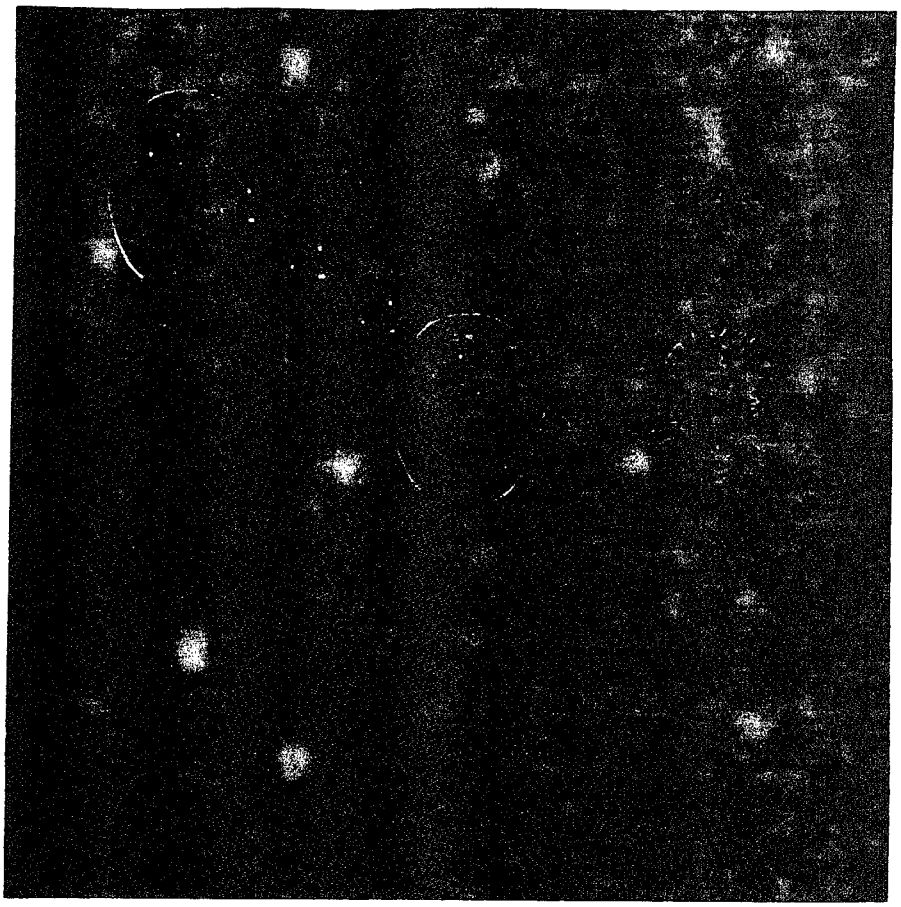


Fig. 1. Illustration of the eigenimage filtering using four spin-echo T2-weighted images (TE/TR = 25-100/2,500 msec) and an inversion recovery image (TE/TI/TR = 20/600/1,500 msec) of a human brain. Original images with the desired and undesired ROIs are shown to the left, the gray matter eigenimage is shown to the right. Signature and weighting vectors are graphed at the bottom.

The solution to the above constrained optimization problem is given by [2]

$$\vec{e} = \vec{d}_o = \vec{d} - \vec{d}_p \tag{4}$$

where $\vec{d}_p$ and $\vec{d}_o$ are the projection of $\vec{d}$ onto the subspace spanned by $\{\vec{u}_i, i = 1, \ldots, m'\}$ and its orthogonal complement, respectively. It can be shown that $\vec{d}_p = U(U^TU)^{-1}U^T\vec{d}$ where U is the $n \times m'$ matrix defined by the undesired signature vectors, i.e., $U = [\vec{u}_1, \vec{u}_2, \ldots, \vec{u}_{m'}]$. The weighting vector $\vec{e}$ can be computed using the numerically-stable Gram-Schmidt orthogonalization. The solution is always nonzero unless $\vec{d}$ is linearly dependent on $\{\vec{u}_i, i = 1, \ldots, m'\}$, which is very unlikely to happen in practice.

As explained in [2] and [3], a major advantage of the eigenimage filter, as compared to other image combination techniques, is its uniqueness in correcting for partial volume averaging effects. In the presence of the multiple interfering features, other filters (e.g., the maximum contrast filter) can neither zero all of the interfering features nor correct for partial volume averaging effects.

For the case of well-defined signature vectors, the SNR of the desired feature in the eigenimage equals the contrast-to-noise ratio (CNR) between the desired feature and any of the interfering features and is expressed by [2], [3]

$$\text{SNR} = \text{CNR} = \frac{||\vec{d}||}{\sigma}\sin(\theta) \tag{5}$$

where $\vec{d}$ is the desired signature vector, σ is the standard deviation of the additive white noise, and θ is the angle between the desired signature vector and its projection onto the undesired subspace (subspace defined by the undesired signature vectors). In the sequel, we will mention the SNR only, since it equals the CNR for the eigenimage.

Equation (5) shows that the SNR of the eigenimage is directly proportional to:

1) The reciprocal $\frac{1}{\sigma}$ of the noise strength (or SNR of the original images);
2) The dissimilarity $||\vec{d}||\sin(\theta)$ between the desired feature and the interfering features.

Therefore, there are two methods for improving the SNR of the eigenimage:

1) Suppressing noise, i.e., improving the SNR of the original images;
2) Optimizing the MRI protocols and pulse sequence parameters, i.e., maximizing $||\vec{d}||\sin(\theta)$.

Improving the SNR of the original images was the subject of [4]. In this paper, we optimize MRI protocols and pulse

sequence parameters to maximize the dissimilarity between the desired and interfering features.

B. Review of Previous MRI Optimization Problems

Optimization of MRI protocols and pulse sequence parameters has been an area of particular interest during the last few years [5]–[18]. In deriving optimal MRI protocols and pulse sequence parameters, several figures-of-merit (FOMs) (or objective functions (OFs)) have been used. These include: (1) Image SNR [5], [6]; (2) image CNR [7]–[11] (sometimes normalized by the square root of the imaging time [12], [13]); (3) signal gradient (parameter sensitivity) [13], [14]; (4) accuracy of the calculated tissue parameters [15]; (5) noise in the synthesized image [16], [17]; and (6) diagnostic test [18].

"Optimizing SNR" maximizes the SNR of a desired tissue in the acquired image. "Optimizing CNR", usually referred to as the *contrast* method, maximizes the difference between the mean values of the image pixel intensities from two specific tissue types. "Optimizing signal gradient" maximizes the sensitivity of the image gray levels to subtle $N(H)$, $T1$, and $T2$ abnormalities within normal tissue. "Optimizing the accuracy of the calculated tissue parameters" or "noise in a synthesized image" minimizes the propagated noise with respect to the calculated values for $N(H)$, $T1$, and $T2$, or to the synthesized image. Finally, "optimizing the diagnostic test" attempts to match the statistical decision boundaries in the signal space to the correct boundaries in the tissue parameter space. Contrast, SNR, and parameter sensitivity are used for optimizing the acquisition of a *single* image, but accuracy of the calculated tissue parameters, noise of the synthesized image, and diagnostic test are used for optimizing the acquisition of *multiple* images, i.e., an *image scene sequence*.

Several methods have been utilized for solving the optimization problems defined by the above FOMs (OFs). Most commonly, SNR [5], CNR [9], noise level [17], or contours of parameter sensitivity [13] are plotted using mathematical models and typical parameter values for normal tissues, e.g., white matter and gray matter. Calculations are performed for multiple combinations of timing intervals. For a given parameter, the location of the maximum SNR, CNR, and sensitivity, or the minimum of noise level, represents the pulse sequence and timing intervals that yield maximum SNR, CNR, and sensitivity to the tissue parameters, or the smallest noise level, respectively. Alternatively, the search for the optimum pulse sequence is transformed into a multidimensional nonlinear optimization problem, and a computer program implementing a numerical method, e.g., a stochastic search algorithm and a modified simplex algorithm, is used to find its solution [12].

All of the above optimization methods require a clinical judgment about which tissues are likely to be affected by the suspected pathologic process, and a knowledge of normal $N(H)$, $T1$, and $T2$ values for those tissues. Optimizing SNR, CNR, calculated tissue parameters, noise level, or diagnostic test requires an accurate estimate of the direction and magnitude of the suspected changes. Parameter sensitivity can be maximized even when the pathologic changes in $N(H)$, $T1$, and $T2$ are unknown or inconsistent.

Not surprisingly, the optimum pulse sequences for different FOMs (OFs) are different. For example, the pulse sequence which optimizes the diagnostic test is different from the one that maximizes CNR [18]. This is because an increase in specificity may require a decrease in CNR of a single image; that is, imaging time may be more wisely spent increasing the specificity of the technique with the addition of different pulse sequences, rather than increasing image CNR by averaging the same pulse sequence.

C. New Optimization Problem

Our problem is different from the previous problems in that our FOM (OF) is not the CNR of a single image but the *normalized* SNR (NSNR = SNR divided by the square root of the acquisition time) of a composite image (eigenimage), generated by a linear combination of several images (an MRI scene sequence) acquired using multiple pulse sequences. It is similar to the optimization problems previously defined for the accuracy of calculated tissue parameters, noise in the synthesized image, and the diagnostic test, in that for all of these problems multiple pulse sequences are used, so that there are several parameters to be optimized.

For optimization, we consider MRI scene sequences generated using four basic MRI protocols and pulse sequences: (1) Multiple spin-echo (MSE); (2) spin-echo (SE); (3) inversion recovery (IR); and (4) gradient-echo (GE). Using: the mathematical expression of the MRI signal from a tissue (an element of a signature vector) [19], [20]; and the intrinsic tissue parameters (N(H), $T1$, and $T2$), found in the MRI literature such as [20], [21] or estimated from the standard $T1$- and $T2$-weighted image sequences, we express the objective function (NSNR of an eigenimage) in terms of the MRI pulse sequence parameters (TE, TI, TR, and flip angle). Table I lists the intrinsic tissue parameters for a quality control (QC) phantom and the normal human brain. The parameters of the QC phantom were estimated using the standard T1 and T2 weighted sequences. For the brain, N(H) and T1 values were obtained from the standard T2–weighted brain images. For the brain, N(H) and T1 values were obtained from the MRI literature [20], [21] and T2 values were estimated from the standard T2–weighted brain images. Note that N(H) is considered as a relative proton density concentration.

The objective function, along with a set of diagnostic or instrumental constraints on the pulse sequence parameters, defines a multidimensional nonlinear constrained optimization problem. Assuming the constraints are consistent, there will be a domain of pulse sequences and pulse sequence parameters from which the optimal parameters are extracted. We translate the constrained optimization problem into a fixed-point problem which we solve using the method of Eaves and Saigal [22] programmed by Saigal [23].

Fixed-point theorems are routinely used to establish the existence of a solution for systems arising in Physics, Mathematical Economics, Operations Research, and the ability to compute these fixed points is an emerging technology. An advantage of this approach is that the underlying functions

TABLE I
INTRINSIC ISSUE PARAMETERS FOR THE QC PHANTOM AND THE NORMAL HUMAN BRAIN

	QC Phantom			Human Brain		
	A	B	C	WM	GM	CSF
N(H)	0.788	0.769	0.786	0.770	0.860	1.000
T1	162.0	1,340.0	300.0	515.0	871.0	1,900.0
T2	72.0	208.0	131.0	56.6	73.3	268.2

TABLE II
TYPICAL IMAGING PARAMETERS FOR MRI BRAIN STUDIES

FOV	ST	MS	NEX	TNAI	MNPS
20 cm	5 mm	192 x 256	1	4 or 5	2

Pulse Sequence Name	Number of Images it Generates
Multiple Spin-Echo (MSE)	2 or 4
Spin Echo (SE)	1
Inversion-Recovery (IR)	1
Gradient-Echo (GE)	1

in optimization). Their use in optimization is recent (see [24]), and their techniques are well suited for problems involving inequalities. In such a case, the methods of unconstrained optimization fail, even when the underlying functions are smooth. Generally, strategies like "active set" or "sequential quadratic programming (SQP)" are adapted in optimization methods to overcome this problem.

The fixed-point formulation we use in this paper takes care of the upper and lower bounds on the variables implicitly, and these constraints are easily incorporated into the formulation. This reduces the size of the problem to be solved, i.e., the number of unknowns to be found. It uses a methodology based on triangulating the space (as in finite element methods) and following a path of solutions to a minimum. This method is ideally suited for nondifferentiable optimization, and thus can handle inequality and equality constraints without explicitly considering the active sets. It is robust in passing the local minima and giving the global minimum, except when there exists a large peak between the local and global minima. Its rate of convergence is quadratic and is thus fast. The optimization technique is demonstrated through its application to the QC phantom and the human brain.

In Section 2, we describe problem formulation and the approach to find a solution. In Section 3, we present theoretical and experimental results for the QC phantom and the human brain. In Section 4, we summarize the research performed and give conclusions, and in the Appendix, we explain some mathematical details for the optimization technique. This paper is an extension of the poster presented at the IEEE Medical Imaging Conference in conjunction with the Nuclear Science Symposium [25].

II. METHODS

A. Problem Formulation

The goal of the optimization is to maximize the SNR of the eigenimage given in (5). Averaging several acquired images (or free induction decay (FID) signals) improves the SNR by a factor of the square root of the number of averaged images (or FID signals), since the additive noise fields are assumed to be independent. Hence, we include the acquisition time (AT) in the formulation of the optimization problem by defining the following NSNR.

$$\text{NSNR} = \frac{\text{SNR}}{\sqrt{\text{AT}}} = \frac{\|\vec{\mathbf{d}}\|}{\sigma\sqrt{\text{AT}}}\sin(\theta) \tag{6}$$

Depending on the application, desired and interfering features are defined, and a single or multiple NSNRs are considered. When using the eigenimage filter to segment an abnormality from normal tissues, the NSNR of the eigenimage, generated by taking the abnormality as the desired feature and normal tissues as the interfering features, is the objective function. When using the eigenimage filter in 3-D visualization, the objective function is the *minimum* of the NSNRs of several eigenimages, each generated by taking one of the tissues as the desired feature and the other tissues as interfering features.

As explained earlier, we solve the optimization problem by translating it to a fixed point problem. This translation requires the gradient of the objective function, as we will explain in Section 2.2. Calculation of the gradient will be much easier if we manipulate and analytically simplify the objective function. For this reason, we use matrix notation and define the following objective function f[1].

$$f = -(\sigma\text{NSNR})^2 = -\frac{\|\vec{\mathbf{d}}\|^2}{\text{AT}}\sin^2(\theta) = -\frac{\|\vec{\mathbf{d}}_o\|^2}{\text{AT}} = -\frac{\vec{\mathbf{d}}_o^T\vec{\mathbf{d}}_o}{\text{AT}} \tag{7}$$

where

$$\vec{\mathbf{d}}_o = \vec{\mathbf{d}} - \vec{\mathbf{d}}_p = [I - U(U^TU)^{-1}U^T]\vec{\mathbf{d}} \tag{8}$$

$$U = [\vec{\mathbf{u}}_1, \vec{\mathbf{u}}_2, \ldots, \vec{\mathbf{u}}_{m'}] = \begin{bmatrix} U_{11} & U_{21} & \cdots & U_{m'1} \\ U_{12} & U_{22} & \cdots & U_{m'2} \\ \vdots & \vdots & \vdots & \vdots \\ U_{1n} & U_{2n} & \cdots & U_{m'n} \end{bmatrix} \tag{9}$$

[1] Minimization of f is equivalent to maximization of NSNR, for a given noise level σ. Without loss of generality, we may also consider $\sigma = 1$, since in deriving MRI signal models, identical constant transmitter and receiver gains are assumed for all of the pulse sequences.

TABLE III
OPTIMUM PULSE SEQUENCE PARAMETERS FOR SEGMENTATION OF ONE FEATURE AND NSs OF THE RESULTANT EIGENVALUES FOR FOUR COMBINATIONS OF MRI SCENE SEQUENCES OF THE QC PHANTOM

PS	DF	TE_{MSE}	TR_{MSE}					NS_A	NS_B	NS_C
4	A	54.05	546.18					24.50	7.59	11.62
MSE	B	94.06	2,430.60					10.60	18.18	9.51
	C	75.64	878.68					20.61	12.57	13.75
PS	DF	TE_{MSE}	TR_{MSE}	TE_{SE}	TE_{SE}			NS_A	NS_B	NS_C
4	A	41.90	732.6	10.0	1.000			119.10	35.30	56.69
MSE	B	20.00	490.9	10.0	2.498			58.79	143.20	57.55
1 SE	C	37.58	786.2	10.0	2.000			83.97	78.13	100.30
PS	DF	TE_{MSE}	TR_{MSE}	TE_{IR}	TI_{IR}	TR_{IR}		NS_A	NS_B	NS_C
4	A	30.52	399.4	12.0	182.2	500.0		227.00	21.93	58.16
MSE	B	19.00	2,500.0	20.0	879.7	2,500.0		41.67	115.20	39.47
1 IR	C	23.52	307.9	12.0	154.0	500.0		132.30	42.45	193.30
PS	DF	TE_{MSE}	TR_{MSE}	TE_{GE}	TI_{GE}	TR_{GE}		NS_A	NS_B	NS_C
4	A	39.17	607.0	5.0	51.09	327.5		150.20	40.23	71.50
MSE	B	20.00	480.0	5.0	58.32	2,000.0		64.05	153.80	62.19
1 GE	C	38.79	607.0	5.0	30.71	383.2		122.00	100.10	124.30

The $n \times 1$ vectors $\vec{\mathbf{d}}_p$ and $\vec{\mathbf{d}}_o$ are the projection of $\vec{\mathbf{d}}$ onto the undesired subspace (the subspace defined by the undesired signature vectors, i.e., the columns of U) and its orthogonal complement, respectively, I is the $n \times n$ identity matrix, and $\{\vec{\mathbf{u}}_i, 1 \le i \le m'\}$ are the interfering signature vectors. Using the relationship between the MRI signal from a tissue (an element of a signature vector), its intrinsic parameters ($N(H)$, $T1$, and $T2$), and MRI protocols and pulse sequence parameters (TE, TI, TR, and flip angle), f in (7) may be expressed as a function of the pulse sequence parameters, i.e., the objective function is $f(\vec{\mathbf{x}})$ where $\vec{\mathbf{x}}$ contains the pulse sequence parameters. Here, we suffice with the above explanation of the objective function. Since each element of $\vec{\mathbf{d}}$ and U is a *complex* function of the above-mentioned parameters (see (45)–(46) for example), an explicit representation of $f(\vec{\mathbf{x}})$ does not fit into the paper.

Constraints: There are two sets of MRI parameters: (1) imaging parameters (matrix size, field of view (FOV), and slice thickness); and (2) pulse sequence parameters (TE, TI, TR, and flip angle). The optimization procedure should include a number of constraints on these parameters. Some of these constraints are specified by the diagnostician and are task dependent, e.g., the minimum required resolution, the total volume to be covered in the examination, and the maximum imaging time. Others are instrumental limitations, e.g., in a sequence of four multiple spin-echo MR images, the echo time (TE) should be greater than or equal to a certain positive number which depends on the imaging equipment. TE should also be less than a quarter of the repetition time (TR). To satisfy these constraints: (1) We fix some of the parameters at typical values for a clinical brain study (see Table II); and (2) for the remaining parameters, we consider lower and upper bounds ($lb_i \le x_i \le ub_i$, $i = 1, \ldots, p$, where x_i is a varying parameter, p is the number of varying parameters, and lb_i and ub_i are lower and upper bounds, respectively), and linear constraints ($g_j(\vec{\mathbf{x}}) \le 0$, $j = 1, \ldots, m$, where m is the number of required relationships among the parameters)[2]. The optimization problem can then be formulated as

$$\text{Minimize} \quad f(\vec{\mathbf{x}})$$
$$\text{subject to} \quad \begin{cases} lb_i \le x_i \le ub_i, & i = 1, \ldots, p \\ g_j(\vec{\mathbf{x}}) \le 0, & j = 1, \ldots, m \end{cases} \tag{10}$$

where $f(\vec{\mathbf{x}})$ was previously defined in (7)–(9), and $\vec{\mathbf{x}} =$ [TE,TI,TR, flip-angle] for example.

For the pulse sequence options and imaging parameters listed in Table II, an image sequence may consist of: (1) Four MSE images; (2) four MSE images and one SE image; (3) four MSE images and one GE image; or (4) four MSE images and one IR image[3]. For each combination, a constrained optimization problem as defined in (10) is solved (using the fixed point approach discussed next) to find the corresponding optimum MRI pulse sequence parameters. Each minimum corresponds to a candidate set of parameters for the final optimal MRI protocol. A comparison of the resulting NSNR (when using these candidate protocols) gives the protocols and pulse sequence parameters that yield the smallest (global) minimum of f (largest (global) maximum of NSNR of the eigenimage).

B. Translation of the New Optimization Problem into a Fixed-Point Problem

A fixed-point problem is defined as follows. Given a point-to-set mapping Γ from $\mathcal{R}^n$ (the n-dimensional Euclidean vector space) into nonempty subsets of $\mathcal{R}^n$, find a point $\vec{\mathbf{x}}$ such that $\vec{\mathbf{x}} \in \Gamma(\vec{\mathbf{x}})$, i.e., $\vec{\mathbf{x}}$ a fixed point of Γ [22], [23]. In particular, if Γ maps points to points, we have $\vec{\mathbf{x}} = \Gamma(\vec{\mathbf{x}})$. In the following, we show how the constrained optimization problem in (10) is translated into a fixed-point problem.

Claim: Define (here we define *row* vectors)

$$x_i^+ = \begin{cases} l_i & \text{if} \quad x_i < lb_i \\ x_i & \text{if} \quad lb_i \le x_i \le ub_i \\ u_i & \text{if} \quad x_i > ub_i \end{cases} \tag{11}$$

[2] For the instrumental limitations given above, we would have the constraints: $\ldots$

[3] The number of images in the sequence is limited by the total acquisition time. Considering the conventional pulse sequences that are available on clinical MRI systems and a reasonable imaging $\ldots$

TABLE IV
OPTIMUM PULSE SEQUENCE PARAMETERS, FOR SEGMENTATION OF ALL THREE FEATURES, AND NSs OF THE RESULTANT EIGENVALUES FOR THREE COMBINATIONS OF MRI SCENE SEQUENCES OF THE QC PHANTOM

PS	DF	TE_{MSE}	TR_{MSE}	TE_{SE}	TR_{SE}		NS_A	NS_B	NS_C
SE	A	31.45	498.7	10.0	1.500		79.09	98.04	80.47
MSE	B	20.00	501.7	10.0	1.498		72.23	109.70	68.29
SE	C	35.16	498.7	10.0	1.500		78.54	93.48	81.12
PS	DF	TE_{MSE}	TR_{MSE}	TE_{IR}	TI_{IR}	TR_{IR}	NS_A	NS_B	NS_C
SE	A	22.94	500.0	20.0	815.3	2.000	61.76	60.74	53.16
MSE	B	19.00	396.1	20.0	635.5	2.000	58.59	90.12	51.91
IR	C	22.13	500.1	20.0	776.7	2.122	60.63	72.99	54.43
PS	DF	TE_{MSE}	TR_{MSE}	TE_{GE}	α_{GE}	TR_{GE}	NS_A	NS_B	NS_C
SE	A	31.17	500	5.0	115.70	2.000	70.36	93..36	71.94
MSE	B	20.00	500	5.0	29.36	500	102.50	133.60	96.85
IR	C	35.33	500	5.0	39.38	500	112.30	111.10	114.20

$$\lambda_j^+ = \max\{0, \lambda_j\} \tag{12}$$

$$\lambda_j^- = \min\{0, \lambda_j\} \tag{13}$$

$$\vec{\mathbf{x}} = [x_1, x_2, \ldots, x_p] \tag{14}$$

$$\vec{\mathbf{x}}^+ = [x_1^+, x_2^+, \ldots, x_p^+] \tag{15}$$

$$\vec{\lambda} = [\lambda_1, \lambda_2, \ldots, \lambda_m] \tag{16}$$

$$\vec{\lambda}^+ = [\lambda_1^+, \lambda_2^+, \ldots, \lambda_m^+] \tag{17}$$

$$\mathbf{g}(\cdot) = [g_1(\cdot), g_2(\cdot), \ldots, g_m(\cdot)] \tag{18}$$

$$\nabla f(\cdot) = \left[\frac{\partial f}{\partial x_1}(\cdot), \frac{\partial f}{\partial x_2}(\cdot), \ldots, \frac{\partial f}{\partial x_p}(\cdot)\right] \tag{19}$$

$$\nabla \mathbf{g}(\cdot) = \begin{bmatrix} \frac{\partial g_1}{\partial x_1}(\cdot) & \frac{\partial g_1}{\partial x_2}(\cdot) & \cdots & \frac{\partial g_1}{\partial x_p}(\cdot) \\ \frac{\partial g_2}{\partial x_1}(\cdot) & \frac{\partial g_2}{\partial x_2}(\cdot) & \cdots & \frac{\partial g_2}{\partial x_p}(\cdot) \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial g_m}{\partial x_1}(\cdot) & \frac{\partial g_m}{\partial x_2}(\cdot) & \cdots & \frac{\partial g_m}{\partial x_p}(\cdot) \end{bmatrix} \tag{20}$$

then solving the following fixed-point problem

$$\begin{cases} \vec{\mathbf{x}}^+ - [\nabla f(\vec{\mathbf{x}}^+) + \vec{\lambda}^+ \nabla \mathbf{g}(\vec{\mathbf{x}}^+)] & = \vec{\mathbf{x}} \\ \vec{\lambda}^+ + \mathbf{g}(\vec{\mathbf{x}}^+) & = \vec{\lambda} \end{cases} \tag{21}$$

yields a solution to (10).

Proof: We show that a solution to (21) is a Karush-Kuhn-Tucker (KKT) point [26] (Section 6.1 in the Appendix shows that for the case of convex f and g_js, the resulting solution yields the minimum of $f(\vec{\mathbf{x}})$). Rewrite (10) as

$$\text{Minimize} \quad f(\vec{\mathbf{x}})$$
$$\text{subject to} \quad \begin{cases} g_j(\vec{\mathbf{x}}) \le 0, & j = 1, \ldots, m \\ x_i \le ub_i, & i = 1, \ldots, p \\ -x_i \le -lb_i, & i = 1, \ldots, p \end{cases} \tag{22}$$

and define the Lagrange multipliers $\vec{\lambda} = [\lambda_1, \ldots, \lambda_m]$, $\vec{\psi} = [\psi_1, \ldots, \psi_p]$, and $\vec{\phi} = [\phi_1, \ldots, \phi_p]$ for the inequalities in (22), respectively. Then a KKT point should satisfy the following conditions [26].

$$\lambda_j, \psi_i, \phi_i \ge 0, \quad j = 1, \ldots, m, i = 1, \ldots, p \tag{23}$$

$$g_j(\vec{\mathbf{x}}) \le 0, \qquad j = 1, \ldots, m \tag{24}$$

$$lb_i \le x_i \le ub_i, \quad i = 1, \ldots, p \tag{25}$$

$$\vec{\lambda}^T \mathbf{g}(\mathbf{x}) = 0 \tag{26}$$

$$\vec{\psi}(\vec{\mathbf{x}} - \vec{\mathbf{ub}})^T = 0, \quad \vec{\mathbf{ub}} = [ub_1, \ldots, ub_p] \tag{27}$$

$$\vec{\phi}(\vec{\mathbf{x}} - \vec{\mathbf{lb}})^T = 0, \quad \vec{\mathbf{lb}} = [lb_1, \ldots, lb_p] \tag{28}$$

$$\nabla f(\vec{\mathbf{x}}) + \vec{\lambda} \nabla \mathbf{g}(\vec{\mathbf{x}}) + \vec{\psi} I - \vec{\phi} I = 0 \tag{29}$$

where I is the $p \times p$ identity matrix. Assume $(\vec{\mathbf{x}}^+, \vec{\lambda}^+)$ is a solution to the fixed-point problem in (21). From (11), (12), (15), and (17) it follows that $\vec{\mathbf{x}}^+$ satisfies (25) and $\vec{\lambda}^+$ satisfies (23). Since $\mathbf{g}(\vec{\mathbf{x}}^+) = \vec{\lambda} - \vec{\lambda}^+ = \vec{\lambda}^-$, from (13) it is clear that (24) is satisfied, also $\vec{\lambda}^+ \mathbf{g}^T(\vec{\mathbf{x}}^+) = \vec{\lambda}^+ (\vec{\lambda}^-)^T = 0$, i.e., (26) is satisfied. From (21) $\nabla f(\vec{\mathbf{x}}^+) + \vec{\lambda}^+ \nabla \mathbf{g}(\vec{\mathbf{x}}^+) = \vec{\mathbf{x}}^+ - \vec{\mathbf{x}} = -\vec{\mathbf{x}}^-$, and for each $\bar{x}_i$ at least one of the following three cases should hold: (1) $\bar{x}_i^+ = \bar{x}_i$, i.e., $lb_i \le \bar{x}_i \le ub_i$; (2) $\bar{x}_i^+ = lb_i$, i.e., $\bar{x}_i \le lb_i$; or (3) $\bar{x}_i^+ = ub_i$, i.e., $\bar{x}_i \ge ub_i$. For (1) $\bar{x}_i^- = 0$ and $\bar{\psi}_i = \bar{\phi}_i = 0$; for (2) $-\bar{x}_i^- \ge 0$, $\bar{\psi}_i = 0$, and $\bar{\phi}_i = -\bar{x}_i^- \ge 0$; and for (3) $-\bar{x}_i^- \le 0$, $\bar{\psi}_i = \bar{x}_i^- \ge 0$, and $\bar{\phi}_i = 0$. Therefore, (23), (27), (28), and (29) are also satisfied. Q.E.D.

Simple Example: In this section, we give a simple example to illustrate the method of translating an optimization problem into a fixed point problem.

Consider the following single variable example.

$$\text{Minimize} \quad f(x) = x^2 - 3x$$
$$\text{subject to} \quad 2 \le x \le 3 \tag{30}$$

According to (11) we define x^+.

$$x^+ = \begin{cases} 2 & \text{if} \quad x < 2 \\ x & \text{if} \quad 2 \le x \le 3 \\ 3 & \text{if} \quad x > 3 \end{cases} \tag{31}$$

Since there is no $g_j(x)$, there is no need for λ_j. The corresponding fixed-point problem according to (21) will be

$$x^+ - \nabla f(x^+) = x \tag{32}$$

But $\nabla f(x) = 2x - 3$ and using (31) we find that

$$x^+ - \nabla f(x^+) = \begin{cases} 1 & \text{if} \quad x < 2 \\ 3 - x & \text{if} \quad 2 \le x \le 3 \\ 0 & \text{if} \quad x > 3 \end{cases} \tag{33}$$

which equals x if and only if $x = 1$. Thus the solution to the fixed-point problem, i.e., the equality in (32), is $x = 1$ and $x^+ = 2$. Note that the solution to the optimization problem is x^+ not x. One can easily show that this solution is correct by plotting $f(x) = x^2 - 3x$ for $2 \le x \le 3$.

Calculation of Gradients: Definition of the fixed-point problem in (21) requires gradients of the objective function and the constraints. The constraints are simple linear functions and their gradients are easily calculated. The objective function is nonlinear and complex, and calculation of its gradient, although straightforward, is somewhat tedious. To proceed with this gradient calculation, define

$$Q = I - U(U^T U)^{-1} U^T \tag{34}$$

then

$$f = -\frac{\vec{\mathbf{d}}^T Q^T Q \vec{\mathbf{d}}}{\mathrm{AT}}. \tag{35}$$

Equation (35) shows that f is a function of Q, $\vec{\mathbf{d}}$, and AT, each of which in turn is a function of the pulse sequence parameters $\vec{\mathbf{x}}$. Hence, using the chain rule we have

$$\nabla_{\mathbf{x}} f = \nabla_Q f \nabla_U Q \nabla_{\mathbf{x}} U + \nabla_{\mathbf{d}} f \nabla_{\mathbf{x}} \vec{\mathbf{d}} + \nabla_{\mathrm{AT}} f \nabla_{\mathbf{x}} \mathrm{AT} \tag{36}$$

where subscripts are variables with respect to which gradients are calculated. Some of these gradients, e.g., $\nabla_{\mathbf{x}} U$ and $\nabla_{\mathbf{x}} \vec{\mathbf{d}}$, are found analytically, others, e.g., $\nabla_Q f$ and $\nabla_U Q$, are found by perturbation method, as explained in Section 6.2 in the Appendix. The results are summarized below.

$$\nabla_Q f = -\frac{2}{\mathrm{AT}}[\mathrm{diag}(Q\vec{\mathbf{d}})\vec{\mathbf{c}}\vec{\mathbf{d}}^T] \tag{37}$$

$$\vec{\mathbf{c}} = [1, 1, \ldots, 1]^T \tag{38}$$

$$\nabla_U Q \Delta U = -\Delta U (U^T U)^{-1} U^T - U(U^T U)^{-1} \Delta U^T + U(U^T U)^{-1}[\Delta U^T U + U^T \Delta U](U^T U)^{-1} U^T \tag{39}$$

$$\nabla_{\mathbf{x}} U = \begin{bmatrix} \frac{\partial U_{11}}{\partial x_1} & \frac{\partial U_{11}}{\partial x_2} & \cdots & \frac{\partial U_{11}}{\partial x_p} \\ \frac{\partial U_{21}}{\partial x_1} & \frac{\partial U_{21}}{\partial x_2} & \cdots & \frac{\partial U_{21}}{\partial x_p} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial U_{nm'}}{\partial x_1} & \frac{\partial U_{nm'}}{\partial x_2} & \cdots & \frac{\partial U_{nm'}}{\partial x_p} \end{bmatrix} \tag{40}$$

$$\nabla_{\mathbf{d}} f = -\frac{2}{\mathrm{AT}} \vec{\mathbf{d}}^T Q^T Q \tag{41}$$

$$\nabla_{\mathbf{x}} \vec{\mathbf{d}} = \begin{bmatrix} \frac{\partial d_1}{\partial x_1} & \frac{\partial d_1}{\partial x_2} & \cdots & \frac{\partial d_1}{\partial x_p} \\ \frac{\partial d_2}{\partial x_1} & \frac{\partial d_2}{\partial x_2} & \cdots & \frac{\partial d_2}{\partial x_p} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial d_n}{\partial x_1} & \frac{\partial d_n}{\partial x_2} & \cdots & \frac{\partial d_n}{\partial x_p} \end{bmatrix} \tag{42}$$

$$\nabla_{\mathrm{AT}} f = \frac{\vec{\mathbf{d}}^T Q^T Q \vec{\mathbf{d}}}{\mathrm{AT}^2} = -\frac{1}{\mathrm{AT}} f \tag{43}$$

$$\nabla_{\mathbf{x}} \mathrm{AT} = [0, \ldots, 0, 1, 0, \ldots, 0, 1] \tag{44}$$

where each one in (44) corresponds to the repetition time (TR) for one of the participating pulse sequences. Again, due to the complexity of the functions we suffice with the above implicit representation of the gradients. Further details for one of the pulse sequence combinations will be given in Section 6.2 in the Appendix.

III. Results

A. Optimal Pulse Sequence Parameters

The optimization procedure described in the Section 2 was applied to MRI scene sequences of the QC phantom and the normal human brain. Intrinsic tissue parameters for the QC phantom were estimated using the standard $T1$- and $T2$-weighted sequences. For the normal brain tissues, N(H) and $T1$ values were obtained from the MRI literature [20], [21] and $T2$ values were estimated from the standard $T2$-weighted brain images. The tissue parameters for the QC phantom and the normal human brain are listed in Table I.

Four combinations of the basic MRI pulse protocols were considered for the optimization. They were: (1) a MSE with four echoes; (2) a MSE with four echoes and a SE; (3) a MSE with four echoes and an IR; and (4) a MSE with four echoes and a GE. Except for the first case, the objective function f was not convex. The optimization program was therefore executed several times using different starting points and different bounds for TR of the participating pulse sequences. Each time a local minimum of the objective function (local maximum of the NSNR of the eigenimage) was found. These local minima were compared to obtain the smallest (global) minimum of the objective function.

Example: As an example, we explain details of the optimization procedure for combination (3). First, the partial derivatives of the signal models, with respect to the pulse sequence parameters, were analytically calculated. Second, a computer program numerically calculated (34)–(44) for each set of the tissue and pulse sequence parameters. This program used the following signal models and their partial derivatives.

$$S^i_{\mathrm{MSE}} = N(H)\left[1 - \left[2\sum_{l=1}^{4}(-1)^l e^{\frac{(2l-1)\mathrm{TE}_{\mathrm{MSE}} - 2\mathrm{TR}_{\mathrm{MSE}}}{2T1}} + e^{\frac{-\mathrm{TR}_{\mathrm{MSE}}}{T1}}\right]\right] e^{\frac{-i\mathrm{TE}_{\mathrm{MSE}}}{T2}}, \qquad 1 \le i \le 4 \tag{45}$$

$$S_{\mathrm{IR}} = N(H)\left[1 - 2e^{\frac{-\mathrm{TI}_{\mathrm{IR}}}{T1}} + 2e^{\frac{-(2\mathrm{TR}_{\mathrm{IR}} - \mathrm{TE}_{\mathrm{IR}})}{2T1}} - e^{\frac{-\mathrm{TR}_{\mathrm{IR}}}{T1}}\right] \times e^{\frac{-\mathrm{TE}_{\mathrm{IR}}}{T2}} \tag{46}$$

$$\frac{\partial S^i_{\mathrm{MSE}}}{\partial \mathrm{TE}_{\mathrm{MSE}}} = N(H)\left[1 - \left[2\sum_{l=1}^{4}(-1)^l e^{\frac{(2l-1)\mathrm{TE}_{\mathrm{MSE}} - 2\mathrm{TR}_{\mathrm{MSE}}}{2T1}} + e^{\frac{-\mathrm{TR}_{\mathrm{MSE}}}{T1}}\right]\right] \frac{-i}{T2} e^{\frac{-i\mathrm{TE}_{\mathrm{MSE}}}{T2}} + N(H)\left[-\sum_{l=1}^{4}(-1)^l \left(\frac{2l-1}{T1}\right) e^{\frac{(2l-1)\mathrm{TE}_{\mathrm{MSE}} - 2\mathrm{TR}_{\mathrm{MSE}}}{2T1}}\right] \times e^{\frac{-i\mathrm{TE}_{\mathrm{MSE}}}{T2}}, \qquad 1 \le i \le 4$$

$$\frac{\partial S^i_{\mathrm{MSE}}}{\partial \mathrm{TR}_{\mathrm{MSE}}} = \frac{N(H)}{T1}\left[2\sum_{l=1}^{4}(-1)^l e^{\frac{-(l-1)\mathrm{TE}_{\mathrm{MSE}}-2\mathrm{TR}_{\mathrm{MSE}}}{2T1}} + e^{\frac{-\mathrm{TR}_{\mathrm{MSE}}}{T1}}\right] e^{\frac{-i\mathrm{TE}_{\mathrm{MSE}}}{T2}}, \quad 1 \le i \le 4 \tag{48}$$

$$\frac{\partial S_{\mathrm{IR}}}{\partial \mathrm{TE}_{\mathrm{IR}}} = N(H)\left[\frac{2}{T2}e^{\frac{-\mathrm{TI}_{\mathrm{IR}}}{T1}} + \left(\frac{1}{T1}-\frac{2}{T2}\right)e^{\frac{-(2\mathrm{TR}_{\mathrm{IR}}-\mathrm{TE}_{\mathrm{IR}})}{2T1}} + \frac{1}{T2}e^{\frac{-\mathrm{TR}_{\mathrm{IR}}}{T1}} - \frac{1}{T2}\right] e^{\frac{-\mathrm{TE}_{\mathrm{IR}}}{T2}} \tag{49}$$

$$\frac{\partial S_{\mathrm{IR}}}{\partial \mathrm{TI}_{\mathrm{IR}}} = N(H)\left[\frac{2}{T1}e^{\frac{-\mathrm{TI}_{\mathrm{IR}}}{T1}}\right] e^{\frac{-\mathrm{TE}_{\mathrm{IR}}}{T2}} \tag{50}$$

$$\frac{\partial S_{\mathrm{IR}}}{\partial \mathrm{TR}_{\mathrm{IR}}} = N(H)\left[\frac{-2}{T1}e^{\frac{-(2\mathrm{TR}_{\mathrm{IR}}-\mathrm{TE}_{\mathrm{IR}})}{2T1}} + \frac{1}{T1}e^{\frac{-\mathrm{TR}_{\mathrm{IR}}}{T1}}\right] e^{\frac{-\mathrm{TE}_{\mathrm{IR}}}{T2}} \tag{51}$$

where S^i_{MSE} represents the signal in the ith image of a four multiple spin-echo sequence with parameters $\mathrm{TE}_{\mathrm{MSE}}$ and $\mathrm{TR}_{\mathrm{MSE}}$, and S_{IR} represents the signal in an inversion recovery image with parameters $\mathrm{TE}_{\mathrm{IR}}$, $\mathrm{TI}_{\mathrm{IR}}$, and $\mathrm{TR}_{\mathrm{IR}}$.

Third, a second computer program used outputs of the first program to implement (11)–(20), with:

$$\vec{\mathbf{lb}} = [lb_1, lb_2, lb_3, lb_4, lb_5] = [19.500k, 12, 100, 500l], k, l = 0, \ldots, 4 \tag{52}$$

$$\vec{\mathbf{ub}} = [ub_1, ub_2, ub_3, ub_4, ub_5] = [100.500(k+1), 20, 500(l+0.9), 500(l+1)], \quad k.l = 0, \ldots, 4 \tag{53}$$

$$\vec{\mathbf{x}} = [x_1, x_2, x_3, x_4, x_5] = [\mathrm{TE}_{\mathrm{MSE}}, \mathrm{TR}_{\mathrm{MSE}}, \mathrm{TE}_{\mathrm{IR}}, \mathrm{TI}_{\mathrm{IR}}, \mathrm{TR}_{\mathrm{IR}}] \tag{54}$$

$$g_1(\vec{\mathbf{x}}) = 4x_1 - x_2 + 50 \tag{55}$$

$$g_2(\vec{\mathbf{x}}) = x_3 + x_4 - x_5 + 50 \tag{56}$$

where the elements of $\vec{\mathbf{lb}}$, $\vec{\mathbf{ub}}$, and $\vec{\mathbf{x}}$ are in milliseconds (msec), and the constraints $g_1(\vec{\mathbf{x}}) \le 0$ and $g_2(\vec{\mathbf{x}}) \le 0$ are similar to those explained in Section 2.1.1. Equation (56) states that $\mathrm{TR}_{\mathrm{MSE}}$ should be greater than or equal to four times $\mathrm{TE}_{\mathrm{MSE}}$, which is the acquisition time for the fourth MSE image, plus 50 msec, which is needed for the sample excitation, signal acquisition, and software management. Equation (56) states that $\mathrm{TR}_{\mathrm{IR}}$ should be greater than or equal to $\mathrm{TE}_{\mathrm{IR}} + \mathrm{TI}_{\mathrm{IR}} + 50$ msec. Again about 50 msec is needed for the sample excitation, signal acquisition, and software management.

Finally, the main subroutine of the fixed point program [23] was modified so that it used the outputs of the above programs and found a solution to (21). For each of the QC phantom and the human brain, and for each of the desired features, the fixed point program was run 25 times ($k.l = 0, \ldots, 4$), each time initializing a locally optimum point. A comparison of the resulting values of the objective function at these locally optimum points yielded the globally optimal parameters for the corresponding pulse sequences.

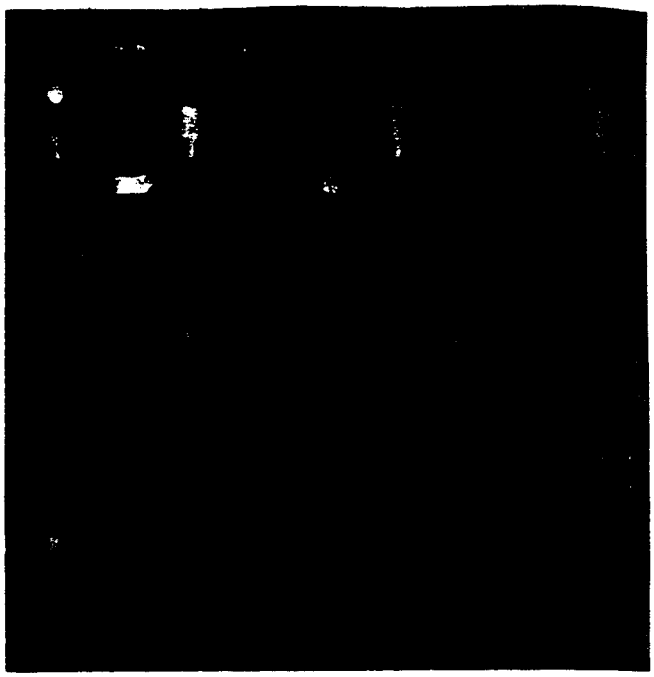


Fig. 2. Original and eigenimages of the QC phantom, using the optimal MRI pulse sequence parameters for a sequence of 4 MSE and 1 SE images. a-e: 4 MSE (TE/TR = 31-124/500 msec) and 1 SE (TE/TR = 10/1,500 msec) images. f-h: Resulting eigenimages for vials A, B, and C, respectively. Orientation of vials A, B, and C in Figs. 2–4 are: A is the circular cross section at 10:00 o'clock position, B is the lengthwise cross sections at 12:00 and 9:00 o'clock positions, and C is the lengthwise cross sections at 3:00.

It should be noted that practical optimization procedures require the functions to be convex to guarantee that the solution found is correct. When this assumption does not hold, one is generally guaranteed only a KKT point, which may even be a saddle-point. This is no different for our formulation, and its convergence to solution can be proved only for the convex case. However, since this formulation can treat the bounds very efficiently, we were able to effectively search for the global minimizer by partitioning the space into sufficiently small "cubical" regions, and searching for a minimizer in each region, and then choosing the best solution found. Although there is no guarantee that the final solution will be the global minimizer, our computation experience appears to support the sufficiency of its accuracy for all practical purposes.

Tables III and IV list the optimum pulse sequence parameters for segmenting one of the features using each of the combinations (1)–(4), for the QC phantom and the human brain, respectively. They also list the σNSNR of the eigenimages generated for each feature (tissue). Tables IV and VI list several optimum pulse sequence parameters for segmenting all of the three features in the scene. These parameters maximize the minimum NSNR of the three eigenimages. They are appropriate for the application of the eigenimage filter to 3-D visualization, since for this application all of the three eigenimages are used and thus the minimum of the three NSNRs determines the quality of the resulting 3-D image.

TABLE V
OPTIMUM PULSE SEQUENCE PARAMETERS, FOR SEGMENTATION OF ONE FEATURE, AND NSs OF THE RESULTANT EIGENIMAGES FOR FOUR COMBINATIONS OF MRI SCENE SEQUENCES OF THE HUMAN BRAIN

PS	DF	TE_{MSE}	TR_{MSE}							
4 MSE	W	43.88	986.60					NS_A	NS_B	NS_C
	G	51.25	1.507.50					9.68	8.31	49.98
	C	84.37	3.085.69					9.00	8.90	68.23
								4.43	5.92	96.28
PS	DF	TE_{MSE}	TR_{MSE}	TE_{SE}	TE_{SE}			NS_A	NS_B	NS_C
4 MSE 1 SE	W	52.56	2.332	10.0	500			46.33	36.62	161.70
	G	52.55	2.499	10.0	500			46.23	36.74	164.40
	C	74.01	2.499	10.0	1.139			30.19	30.43	182.50
PS	DF	TE_{MSE}	TR_{MSE}	TE_{IR}	TI_{IR}	TR_{IR}		NS_A	NS_B	NS_C
4 MSE 1 IR	W	31.34	1.000	12.0	631.2	2.336		107.70	64.48	95.86
	G	19.00	1.059	12.0	555.3	2.500		79.36	105.70	95.70
	C	57.64	2.500	12.0	360.8	1.000		53.53	54.77	198.60
PS	DF	TE_{MSE}	TR_{MSE}	TE_{GE}	α_{GE}	TR_{GE}		NS_A	NS_B	NS_C
4 MSE 1 GE	W	54.84	2.207	5.0	104.64	456.1		48.05	37.97	165.30
	G	54.60	2.424	5.0	104.86	399.6		47.37	38.38	171.20
	C	73.03	2.500	5.0	60.98	82.0		33.35	32.21	183.70

The following notation is used in Tables III–VI:

PS	:	pulse sequence
DF	:	desired feature
TE_{MSE}	:	echo time (TE), in msec, for the MSE pulse sequence
TR_{MSE}	:	repetition time (TR), in msec, for the MSE pulse sequence
TE_{SE}	:	echo time (TE), in msec, for the SE pulse sequence
TR_{SE}	:	repetition time (TR), in msec, for the SE pulse sequence
TE_{IR}	:	echo time (TE), in msec, for the IR pulse sequence
TI_{IR}	:	inversion time (TI), in msec, for the IR pulse sequence
TR_{IR}	:	repetition time (TR), in msec, for the IR pulse sequence
TE_{GE}	:	echo time (TE), in msec, for the GE pulse sequence
α_{GE}	:	pulse flip angle (α), in degrees, for the GE pulse sequence
TR_{GE}	:	repetition time (TR), in msec, for the GE pulse sequence
A, B, C	:	different materials (features) in the QC phantom (see captions of Fig. 2)
WM, GM, CSF	:	brain tissues: white matter, gray matter, cerebrospinal fluid
NS_q	:	σNSNR of the feature q in the eigenimage

The optimization results listed in Tables III-VI suggest that the addition of either a SE, IR, or GE image to the sequence of four MSE images significantly improves the quality (NSNR) of the eigenimage. This result may be interpreted as follows. At least three intrinsic tissue parameters influence the MRI signal, but only two parameters are adjustable in a MSE pulse sequence. With a second pulse sequence, at least four parameters can be adjusted, hence, there are enough degrees of freedom for generating high NSNR eigenimages. Tables III—VI also suggest that a sequence of four MSE and one GE

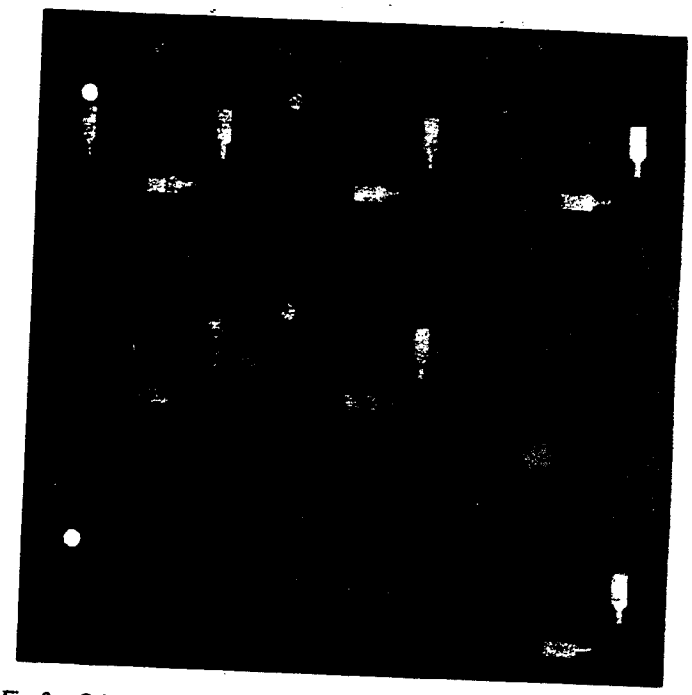


Fig. 3. Original and eigenimages of the QC phantom, using the optimal MRI pulse sequence parameters for a sequence of 4 MSE and 1 IR images. (a)–(e) 4 MSE (TE/TR = 22–88/500 msec) and 1 IR (TE/TI/TR = 20/777/2,122 msec) images. (f)–(g) Resulting eigenimages for vials *A*, *B*, and *C*.

with optimal pulse sequence parameters, should generate phantom eigenimages with largest NSNRs. Similarly, a sequence of four MSE and an IR, with optimal pulse sequence parameters, should generate brain eigenimage with largest NSNRs.

C. Experiments

Several experiments were performed to evaluate the mathematical predictions. The QC phantom was imaged using the optimal pulse sequence parameters for segmenting all of the three features in the scene (i.e., the one that maximizes the minimum NSNR for the three eigenimages), for each pulse sequence combination given in Table IV. It was also imaged using the conventional pulse sequence parameters for

TABLE VI
OPTIMUM PULSE SEQUENCE PARAMETERS, FOR SEGMENTATION OF ALL THREE FEATURES, AND NSs OF THE RESULTANT EIGENIMAGES FOR THREE COMBINATIONS OF MRI SCENE SEQUENCES OF THE HUMAN BRAIN

PS	DF	TE_{MSE}	TR_{MSE}	TE_{SE}	TR_{SE}		NS_A	NS_B	NS_C
4	W	52.56	2,332	10.0	500		46.33	36.62	161.70
MSE	G	52.55	2,499	10.0	500		46.23	36.74	164.40
1 SE	C	68.29	2,499	10.0	500		44.73	35.24	171.80
PS	DF	TE_{MSE}	TR_{MSE}	TE_{IR}	TI_{IR}	TR_{IR}	NS_A	NS_B	NS_C
4	W	28.70	2,000	12.0	632.9	2,500	101.50	72.62	133.10
MSE	G	19.00	1,500	12.0	519.0	2,500	90.26	94.19	115.10
1 IR	C	51.35	2,500	20.0	519.0	2,500	74.68	74.85	187.40
PS	DF	TE_{MSE}	TR_{MSE}	TE_{GE}	α_{GE}	TR_{GE}	NS_A	NS_B	NS_C
4	W	54.21	2,236	5.0	107.42	500	47.94	37.77	164.00
MSE	G	53.51	2,467	5.0	111.95	500	47.52	38.06	168.00
1 GE	C	67.88	2,500	5.0	126.52	500	44.47	36.42	175.20

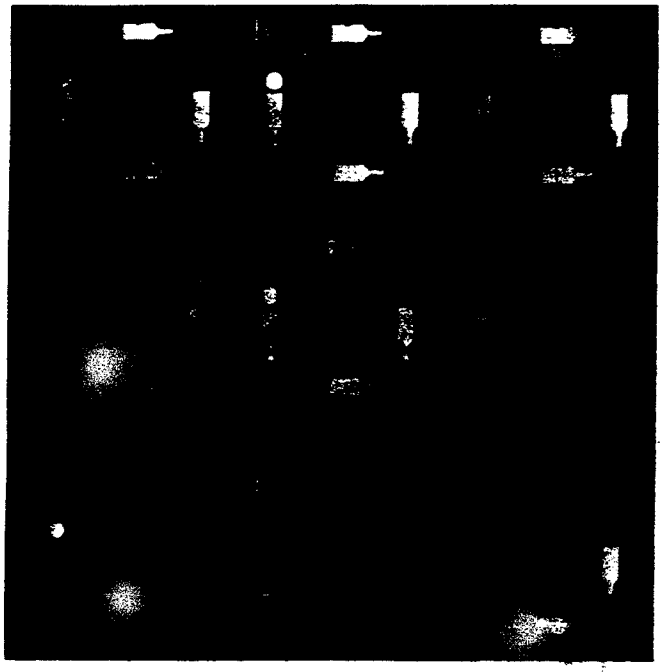


Fig. 4. Original and eigenimages of the QC phantom, using the optimal MRI pulse sequence parameters for a sequence of four MSE and one GE images. (a)–(e) four MSE (TE/TR = 35–140/2,500 msec) and one GE (TE/TR = 5/500 msec, α = 40 deg) images. (f)–(h) Resulting eigenimages for vials A, B, and C, respectively.

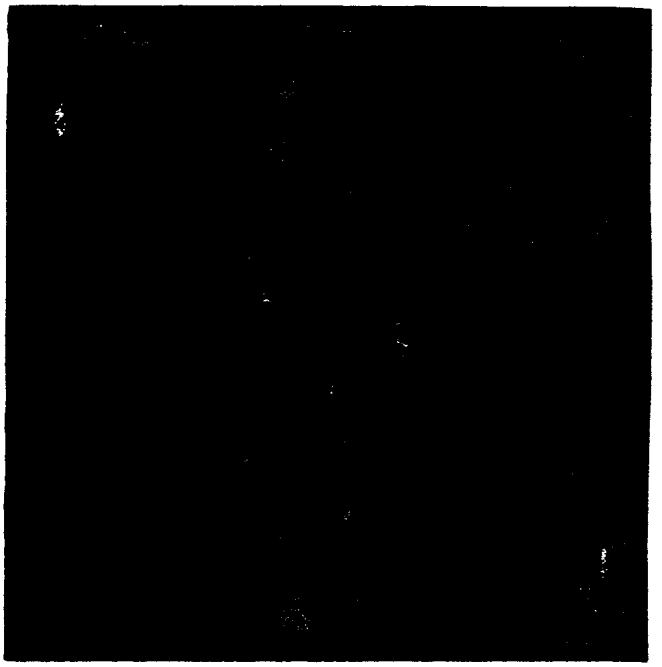


Fig. 5. Original and eigenimages of the QC phantom, using the conventional MRI pulse sequence and parameters. (a)–(e) 4 MSE (TE/TR = 25 – 100/2,500 msec) and 1 SE (TE/TR = 20/500 msec) images. (f)–(h) Resulting eigenimages for vials *A*, *B*, and *C*, respectively.

parameters are given in the captions of Table VII.). For each experiment, three eigenimages were generated and their NSNRs were estimated. Original and eigenimages of the QC phantom are shown in Figs. 2–5. The mathematical and experimental values for the NSNRs of the resulting eigenimages are compared in Table VII (The mathematical values are obtained by dividing σNSNRs in Table IV by an estimate of σ.).

Nine human volunteers were imaged using both the conventional and the optimal pulse sequence parameters for segmenting all of the three brain tissues (The MRI parameters are given in the captions of Table VIII.). For each experiment, three eigenimages were generated and their NSNRs were estimated. Table VIII compares the resulting NSNRs of the nine volunteers' eigenimages using conventional MRI pulse sequence parameters for brain studies with those using the optimal MRI pulse sequence parameters. The ratio of the NSNR of the eigenimage generated using optimal parameters to that of conventional parameters illustrates the improvement attained as a result of the optimization. This ratio is 1.26 ± 0.31 for white matter, 2.12 ± 0.64 for gray matter, and 1.13 ± 0.21 for cerebrospinal fluid (CSF). Considering the improvement for the gray matter eigenimage (since it has usually a smaller NSNR than the white matter and CSF eigenimages), the optimal pulse sequence almost doubles the smallest NSNR of the brain eigenimages. This indicates that using the optimal protocol and pulse sequence parameters, the imaging time for the brain eigenimage filtering may be reduced by 75% if limited number of slices (less than 10) is going to be imaged. For entire brain imaging this saving may be less since more than one acquisition is needed to cover the entire volume. As an example, the original and eigenimages of a volunteer's brain[4] (volunteer # 7) are shown in Figs. 6 and 7.

[4]Note that segmentation of the skull (or muscles and skin) is not usually (and thus in this paper) the purpose of the eigenimage filtering. Segmenting the

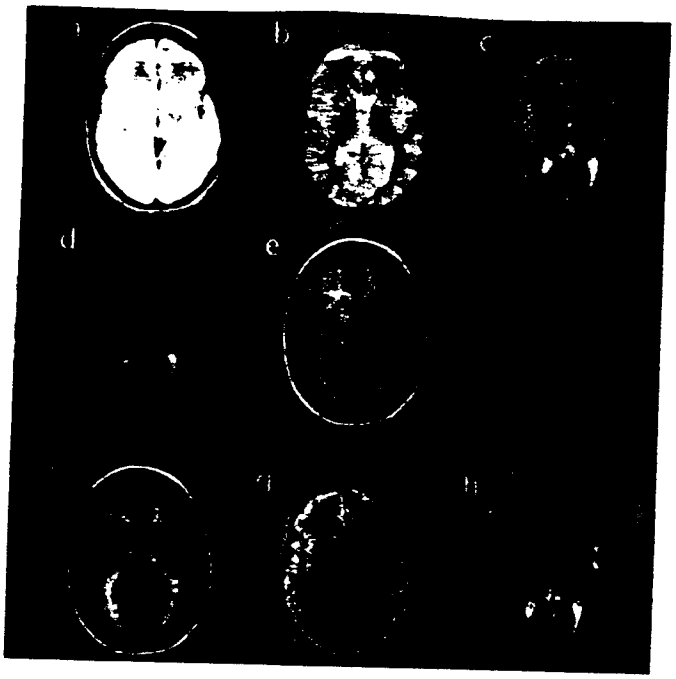


Fig. 6. Original and eigenimages of a volunteer's brain (volunteer #7) using the conventional MRI pulse sequence and parameters. (a)–(e) four MSE (TE/TR = 25 – 100/2,500 msec) and one SE (TE/TR = 20/500 msec). (f)–(h) Resulting eigenimages for white matter, gray matter, and CSF, respectively.

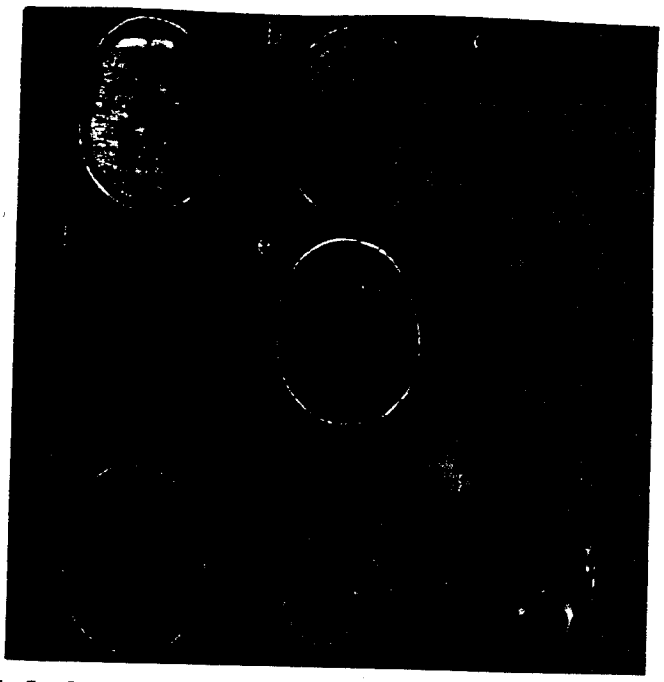


Fig. 7. Original and eigenimages of the volunteer's brain (volunteer #7) using the optimal MRI pulse sequence and parameters. (a)–(e) four MSE (TE/TR = 19–76/ 1,500 msec) and one IR (TE/TI/TR = 12/519/2000 msec). (f)–(h) Resulting eigenimages for white matter, gray matter, and CSF, respectively.

The improvement in the quality of the eigenimages, as a result of optimization, is clearly seen in Fig. 7, even better than what is inferred from NSNRs in Table VIII. The mathematical and experimental values for the NSNRs of the brain eigenimages are compared in Table IX.

Using the optimal pulse sequence parameters for a sequence of only four MSE images, similar experiments for the QC phantom and the human brain were performed (Results are not shown here for conciseness.). It was noticed that the resulting eigenimages in both cases were superior to those generated using conventional values for the pulse sequence parameters.

All of the experiments confirmed that an improvement can be obtained by using the optimal parameters. There were, however, some differences between the mathematical and experimental results. We attribute these differences to the model inaccuracies and the error in estimating NSNRs and tissue parameters (e.g., as a result of difficulty in selecting pure regions of interest) as well as the person-to-person variation of the tissue parameters.

TABLE VII
COMPARISON OF THE NSNRS OF THE QC PHANTOM EIGENIMAGES, USING FOUR COMBINATIONS OF MRI PULSE SEQUENCES

		$NSNR_A$	$NSNR_B$	$NSNR_C$
a	Mathematical	9.89	12.26	10.06
	Experimental	13.83	9.34	9.03
b	Mathematical	7.58	9.12	6.80
	Experimental	6.13	9.28	8.10
c	Mathematical	14.04	13.89	14.28
	Experimental	20.66	11.94	9.60
d	Mathematical	5.35	7.72	3.96
	Experimental	5.12	9.92	3.47

IV. SUMMARY AND DISCUSSION

We optimized MRI protocols and pulse sequence parameters for the eigenimage filtering. We formulated the maximization of the NSNR of the eigenimage as a multidimensional nonlinear constrained optimization problem, which we solved by the fixed-point approach. The fixed-point approach was used because of the ease with which it handles the upper and lower bounds.

We found the mathematical predictions for the optimal MRI parameters for the QC phantom and the human brain. We then performed several experiments to evaluate the mathematical predictions. These experiments confirmed that an improvement can always be obtained by using the optimal parameters. We found that among the basic MRI protocols currently available on clinical systems, a sequence of four MSE and an IR images, using the optimal pulse sequence parameters, would generate excellent (i.e., with high SNR) brain eigenimages.

Maximization of the NSNR of the eigenimage corresponds to an appropriate splitting of the time between the participating pulse sequences to maximize the "distinguishability" or "dissimilarity" of the desired feature from the interfering features in the n-dimensional feature space spanned by the pixel vectors defined from the image sequence. An intuitive explanation for the optimality of the MSE + IR combination is as follows. The MRI signal in magnitude reconstructed images is always positive, hence the most dissimilar signature vectors correspond to the case with a zero element for

brain tissues (intracranial tissues) from the muscles and skin, which are outside the skull, is simple, since there is a dark (very low signal) area, corresponding to the skull (bone), between the two. We have a region growing algorithm for this segmentation if necessary. In Fig. 7, the muscles and skin were not removed, since they serve as reference structures without interfering with the desired feature.

TABLE VIII
EXPERIMENTAL NSNRs OF THE BRAIN EIGENIMAGES AND THE IMPROVEMENT RATIO FOR NINE VOLUNTEERS. CONSIDERING CONVENTIONAL MRI PARAMETERS (A MSE WITH TE/TR=25/2.500 MSEC AND A SE WITH TE/TR=20/500 MSEC) AND THE OPTIMAL MRI PARAMETERS (A MSE WITH TE/TR=19/1.500 MSEC AND AN IR WITH TE/TI/TR=12/519/2.000 MSEC)

		$NSNR_W$	$NSNR_G$	$NSNR_I$
Volunteer #1	con	3.73	2.49	6.59
	opt	6.21	7.18	9.57
	ratio	1.66	2.88	1.45
Volunteer #2	con	3.80	3.27	7.52
	opt	5.53	4.75	7.93
	ratio	1.45	1.45	1.05
Volunteer #3	con	4.60	4.68	9.74
	opt	7.32	9.06	11.80
	ratio	1.59	1.93	1.21
Volunteer #4	con	4.68	4.12	9.75
	opt	7.68	7.80	10.28
	ratio	1.64	1.89	1.05
Volunteer #5	con	4.86	4.82	7.24
	opt	4.77	6.88	8.71
	ratio	0.98	1.43	1.20
Volunteer #6	con	5.76	2.93	18.00
	opt	6.00	7.31	12.51
	ratio	1.04	2.49	0.69
Volunteer #7	con	5.75	3.02	11.51
	opt	6.38	9.75	11.30
	ratio	1.11	3.23	0.98
Volunteer #8	con	7.92	3.89	11.65
	opt	6.13	5.31	13.28
	ratio	0.77	1.40	1.14
Volunteer #9	con	4.50	2.62	9.88
	opt	4.75	6.39	13.69
	ratio	1.05	2.43	1.38
Mean Values	con	5.06	3.54	10.20
	opt	6.08	7.16	11.01
	ratio	1.26	2.12	1.13
Standard Deviation	con	1.21	0.82	3.23
	opt	0.94	1.52	1.90
	ratio	0.31	0.64	0.21

vector. The IR pulse sequence is unique in that it can generate an image in which the average gray levels from a tissue is close to zero (note that MSE is included in all of the combinations). An appropriate selection of the pulse sequence parameters for this protocol (MSE + IR) is therefore expected to generate the best NSNR in the eigenimage.

Although we considered normal brain tissues for the feasibility studies in this research, the optimization method will be applicable to abnormal tissues, as long as fair estimates of the abnormal tissue parameters are available. These estimates can usually be obtained from the literature. If they are not found in the literature, they need to be estimated once. It would be possible to use the optimal parameters to image patients with similar conditions. For example, optimal protocol for the eigenimage filtering of stoke patients can be established by estimating intrinsic parameters for the stroke lesion and the normal brain tissues from typical stroke patients.

It should be noted that the derivation of the eigenimage filter (and, in fact, all of the other image combination techniques [3]) is based on a homogeneity assumption, that is the average value of the image gray levels in a pure ROI (i.e., without including any partial volume pixels) coincides with the MRI signal for the corresponding tissue (i.e., if there was no noise or artifact in the image). In other words, a single signature vector is enough for representing a tissue type. The homogeneity assumption may not hold for certain types of lesions, which we refer to as multiple-zone lesions, since they cannot be characterized by a single signature vector. In this situation, using the average of the signature vectors for different zones of the lesion as the desired signature vector and the surrounding normal tissues as the interfering features would generate an eigenimage in which the lesion is segmented from the normal tissues. Having different average gray levels for different zones of the lesion makes it possible to distinguish them from each other. One approach for further processing would be to look at the histogram of the eigenimage which may allow segmentation of different zones by selecting gray level cut-offs at the histogram valleys. A second approach would be to segment the entire lesion (and generate the corresponding mask) by appropriately thresholding the eigenimage, and then subimaging the original images with the mask, and finally running the eigenimage filter by defining one zone of the lesion as the desired feature and the other zones as the interfering features. For the optimization, one may optimize the segmentation of the normal tissues from one another, or the segmentation of the average lesion from the normal tissues.

Note that the time needed for the estimation of the tissue parameters is not related to the final optimal protocol. The

TABLE IX
COMPARISON OF THE NSNRS OF THE BRAIN EIGENIMAGE USING: (A) CONVENTIONAL MRI PULSE SEQUENCE AND PARAMETERS, FOUR MSE (TE/TR=25–100/2,500 MSEC) AND 1 SE (TE/TR=20/500 MSEC); AND (B) OPTIMAL PULSE SEQUENCE AND PARAMETERS, 4 MSE (TE/TR=19/1,500 MSEC) AND 1 IR (TE/TI/TR=12/519/2,000 MSEC)

		$NSNR_W$	$NSNR_G$	$NSNR_C$
a	Mathematical	4.07	3.24	11.93
	Experimental	5.06	3.54	10.20
b	Mathematical	6.83	7.13	8.72
	Experimental	6.08	7.15	11.01

reason being the use of standard protocols for the estimation of the tissue parameters. We are not optimizing these standard protocols.

The sensitivity of the objective function to the tissue parameters can be estimated by calculating partial derivatives of the objective function with respect to each of the tissue parameters. It can be empirically found by calculating the difference between the theoretically expected value of the objective function with the experimental one. These were considered beyond the scope of this paper in order to keep the length of the paper reasonable. In addition to the person-to-person and the section-to-section variation of the tissue parameters, there are other inaccuracies, e.g., that of the MRI signal models. The practical issue is the over-all effect of these imperfections on the final results, which has been examined for the QC phantom and normal brain. Tables VII and IX compare the mathematical and experimental values for the NSNRs of the resulting eigenimages, and illustrate the overall sensitivity of the objective function to all of the imperfections. Table VII shows four combinations of MROI pulse sequences (three optimal and the conventional): (a) 4 MSE (TE/TR=31.45/498.7 msec) and 1 SE (TE/TR=10.0/1,500 msec); (b) 4 mse (TE/TR=22.13/500.1 msec) and 1 IR (TE/TR=20.0/776.7/2,122 msec); (c) 4 MSE (TE/TR=35.33/500 msec) and 1 GE (TE/TR=5.0/500 msec, α=39.38 deg); (d) 4 MSE (TE/TR=25/2,500 msec) and 1 SE (TE/TR=20/500 msec). Furthermore, the experimental results of imaging nine volunteers in Table VIII show that the use of the optimal pulse sequence has improved the NSNR for all of the brain eigenimages. In Table VIII, conventional MRI parameters are an MSE with TE/TR=25/2,500 msec and an SE WITH TE/TR=20/500 msec and the optimal MRI parameters are an mse with TE/TR=19/1,500 msec and an IR with TE/TI/TR=12/519/2,000 msec. An underlying reason for the ability to obtain this improvement is the fact that the eigenimage filter does not propagate the aforementioned imperfections, since it finds and uses the signature vectors from each slice of the acquired data and not from the MRI signal models.

An important result of our investigation was that on average the optimal pulse sequence would almost double the smallest NSNR of the brain eigenimages, as compared to the conventional brain protocol. This indicates that using the optimal protocol and pulse sequence parameters would reduce the imaging time for the eigenimage filtering of brain studies up to 75%.

The optimization technique can be modified to find the optimum MRI protocols and pulse sequence parameters for filters (transformations) other than the eigenimage filter. For example, using the analytical expressions for the SNR or CNR of the transformed images given in [3], similar optimization problems can be formulated and solved, yielding the optimum MRI protocols and pulse sequence parameters for the corresponding filters.

Finally, the optimization technique can be extended so that it considers the recently developed pulse sequences for fast MRI scans such as turbo flash, fast spin-echo, and even echo planar. This extension will make it possible to optimally utilize a variety of the newest MRI pulse sequences, in conjunction with eigenimage filtering, to assist radiologists in extracting the maximum amount of information from an increasingly larger data set.

APPENDIX

A. Minimizing Convex Functions with Convex Constraints

In the following we show that for a convex objective function and a set of convex constraints, the solution to the fixed-point problem in (21) is the minimizer of the objective function.

Assume $(\vec{\mathbf{x}}^+, \vec{\lambda}^+)$ is the fixed point of (21), and $\vec{\mathbf{y}}$ is a feasible point, i.e., it satisfies all of the constraints. We prove $\vec{\mathbf{x}}^+$ is the minimizer by showing that $f(\vec{\mathbf{x}}^+) \le f(\vec{\mathbf{y}})$. From the convexity assumption, we have

$$f(\vec{\mathbf{y}}) \ge f(\vec{\mathbf{x}}^+) + \nabla f(\vec{\mathbf{x}}^+)(\vec{\mathbf{y}} - \vec{\mathbf{x}}^+)^T \tag{57}$$

$$g_j(\vec{\mathbf{y}}) \ge g_j(\vec{\mathbf{x}}^+) + \nabla g_j(\vec{\mathbf{x}}^+)(\vec{\mathbf{y}} - \vec{\mathbf{x}}^+)^T, \quad j = 1, \ldots, m. \tag{58}$$

Since $\lambda_j^+ \ge 0$ and $g_j(\vec{\mathbf{y}}) \le 0$, $\vec{\lambda}^+ \mathbf{g}^T(\vec{\mathbf{y}}) \le 0$ and we can write

$$f(\vec{\mathbf{y}}) \ge f(\vec{\mathbf{y}}) + \vec{\lambda}^+ \mathbf{g}^T(\vec{\mathbf{y}}). \tag{59}$$

Substituting the right-hand sides of (57) and (58) into the right-hand side of (59), we obtain

$$\begin{aligned} f(\vec{\mathbf{y}}) &\ge f(\vec{\mathbf{x}}^+) + \nabla f(\vec{\mathbf{x}}^+)(\vec{\mathbf{y}} - \vec{\mathbf{x}}^+)^T + \vec{\lambda}^+ [\mathbf{g}^T(\vec{\mathbf{x}}^+) \\ &\quad + \nabla \mathbf{g}(\vec{\mathbf{x}}^+)(\vec{\mathbf{y}} - \vec{\mathbf{x}}^+)^T] \\ &\ge f(\vec{\mathbf{x}}^+) + [\nabla f(\vec{\mathbf{x}}^+) + \vec{\lambda}^+ \nabla \mathbf{g}(\vec{\mathbf{x}}^+)](\vec{\mathbf{y}} - \vec{\mathbf{x}}^+)^T \\ &\ge f(\vec{\mathbf{x}}^+). \end{aligned} \tag{60}$$

In the above we used (26) and

$$[\nabla f(\vec{\mathbf{x}}^+) + \vec{\lambda}^+ \nabla \mathbf{g}(\vec{\mathbf{x}}^+)](\vec{\mathbf{y}} - \vec{\mathbf{x}}^+)^T \ge 0. \tag{61}$$

Inequality (61) holds because from (21) $\nabla f(\vec{\mathbf{x}}^+) + \vec{\lambda}^+ \nabla g$ $(\vec{\mathbf{x}}^- = \vec{\mathbf{x}}^+ - \vec{\mathbf{x}} = -\vec{\mathbf{x}}^-$ and for

the following three cases should hold: (1) $\bar{x}_i^+ = \bar{x}_i$, i.e., $lb_i \le \bar{x}_i \le ub_i$; (2) $\bar{x}_i^- = lb_i$, i.e., $\bar{x}_i \le lb_i$; or (3) $\bar{x}_i^+ = ub_i$, i.e., $\bar{x}_i \ge ub_i$. For (1) $\bar{x}_i^- = 0$; for (2) $-\bar{x}_i^- \ge 0$ and $y_i \ge lb_i = x_i^-$, i.e., $-\bar{x}_i^-(y_i - x_i^-) \ge 0$; and for (3) $-\bar{x}_i^- \le 0$ and $y_i \le ub_i = x_i^-$, i.e., $-\bar{x}_i^-(y_i - x_i^-) \ge 0$. Q.E.D.

B. Gradient Calculation by Perturbation Method

Let us start with the calculation of $\nabla_U Q$.

$$\begin{aligned} Q(U+\Delta U) &= I - (U+\Delta U)[(U+\Delta U)^T(U+\Delta U)]^{-1} \\ &\quad (U+\Delta U)^T \\ &= I - (U+\Delta U)[U^TU + \Delta U^TU \\ &\quad + U^T\Delta U + o(||\Delta U||^2)]^{-1}(U+\Delta U)^T \\ &= I - (U+\Delta U)[(U^TU)^{-1} - (U^TU)^{-1} \\ &\quad [\Delta U^TU + U^T\Delta U](U^TU)^{-1}] \\ &\quad (U+\Delta U)^T + o(||\Delta U||^2) \\ &= [I - U(U^TU)^{-1}U^T] - \Delta U(U^TU)^{-1}U^T \\ &\quad - U(U^TU)^{-1}\Delta U^T + U(U^TU)^{-1} \\ &\quad \times [\Delta U^TU + U^T\Delta U] \\ &\quad (U^TU)^{-1}U^T + o(||\Delta U||^2) \end{aligned} \tag{62}$$

Hence

$$\begin{aligned} \nabla_U Q\Delta U' = &-\Delta U(U^TU)^{-1}U^T - U(U^TU)^{-1}\Delta U^T \\ &+ U(U^TU)^{-1}[\Delta U^TU + U^T\Delta U](U^TU)^{-1}U^T \end{aligned} \tag{63}$$

Here $\Delta U' = [\Delta U_{11}, \Delta U_{21}, \ldots, \Delta U_{m'n}]^T$ is a column vector generated by sweeping ΔU row-by-row, and $\nabla_U Q$ is a $n^2 \times nm'$ matrix whose (rs)th column is found by substituting ΔU by H_{rs}^U. The $n \times m'$ matrix $H_{rs}^U(i,j), 1 \le i \le n, 1 \le j \le m'$ is defined as

$$H_{rs}^U(i,j) = \begin{cases} 1 & \text{if } i = r \text{ and } j = s \\ 0 & \text{otherwise.} \end{cases} \tag{64}$$

Sweeping over $1 \le r \le n$ and $1 \le s \le m'$ generates all nm' columns of $\nabla_U Q$. Similarly,

$$\begin{aligned} f(Q+\Delta Q) &= -\frac{1}{AT}[d^T(Q+\Delta Q)^T(Q+\Delta Q)d] \\ &= -\frac{1}{AT}[d^T(Q^TQ + \Delta Q^TQ + Q^T\Delta Q + \Delta Q^T\Delta Q)d] \\ &= -\frac{1}{AT}[d^TQ^TQd + d^T(\Delta Q^TQ + Q^T\Delta Q)d \\ &\quad + o(||\Delta Q||^2)]. \end{aligned} \tag{65}$$

Thus

$$\nabla_Q f\Delta Q' = -\frac{1}{AT}[d^T(\Delta Q^TQ + Q^T\Delta Q)d]. \tag{66}$$

Again $\Delta Q'$ is generated by sweeping ΔQ row-by-row. Using H_{rs}^Q, as defined below, columns (elements) of $\nabla_Q f$ are found (note that $\nabla_Q f$ is a $1 \times n^2$ vector).

$$H_{rs}^Q(i,j) = \begin{cases} 1 & \text{if } i = r \text{ and } j = s \\ 0 & \text{otherwise} \end{cases} \tag{67}$$

where $1 \le i, j \le n$.

ACKNOWLEDGMENT

The authors would like to thank Donald Peck, Steve Ramsey, Lucie Bower, and Linda LeBlanc from the Medical Image Analysis Laboratory of Henry Ford Hospital for their help with programming, data collection, and analysis.

REFERENCES

[1] J. P. Windham, M. A. Abd-Allah, D. A. Reimann, J. W. Froelich, and A. M. Haggar, "Eigenimage filtering in MR imaging," *J. Comput. Ass. Tomography*, vol. 12 no. 1, pp. 1–9, 1988.

[2] H. Soltanian-Zadeh, J. P. Windham, and A. E. Yagle, "Optimal transformation for correcting partial volume averaging effects in magnetic resonance imaging," IEEE Trans. Nuc. Sci., July/Aug. 1993.

[3] H. Soltanian-Zadeh, J. P. Windham, D. J. Peck, and A. E. Yagle, "A comparative analysis of several transformations for enhancement and segmentation of magnetic resonance image scene sequences," *IEEE Trans. Med. Imag.*, vol. 11, no. 3, pp. 2–18, Sept. 1992.

[4] H. Soltanian-Zadeh, J. P. Windham, and A. E. Yagle: "Magnetic resonance image restoration using a new multi-dimensional non-linear edge-preserving filter." Submittd to *IEEE Trans. Imag. Proc.*.

[5] R. E. Hendrick, F. D. Newman, and W. R. Hendee, "MR Imaging technology: Maximizing the signal-to-noise ratio from a single tissue," *Radiology*, vol. 13, no. 3, pp. 749–752, 1985.

[6] C. B. Ahn, S. Y. Lee, O. Nalcioglu, and Z. H. Cho, "An improved nuclear magnetic resonance diffusion coefficient imaging method using an optimized pulse sequence," *Med. Phys.*, vol. 13, no. 6, pp. 789–793, Nov./Dec. 1986.

[7] W. L. Greif, R. B. Buxton, R. B. Lauffer et. al., "Pulse sequence optimization for MR imaging using a paramagnetic hepatobiliary contrast agent," *Radiology*, vol. 157, pp. 461–466, 1985.

[8] C. N. de Graaf and C. J. G. Bakker, "Simulation procedure to determine nuclear magnetic resonance imaging pulse sequence parameters for optimal tissue contrast," *J. Nuclear Med.*, vol. 27, no. 2, pp. 281–286, Feb. 1986.

[9] R. M. Henkelman et. al., "Optimal pulse sequence for imaging hepatic metastases," *Radiology*, vol. 161, pp. 727–734, 1986.

[10] M. R. Paling et. al., "Liver metastases: Optimization of MR imaging pulse sequences at 1.0 T," *Radiology*, vol. 167, pp. 695–699, 1988.

[11] C. J. Fretz et. al., "Superparamagnetic iron oxide-enhanced MR imaging: Pulse sequence optimization for detection of liver cancer," *Radiology*, vol. 172, pp. 393–397, 1989.

[12] W. Dreher and P. Bornert "Pulse sequence and parameter choice in NMR imaging as a problem of constrained multidimensional nonlinear optimization," *Magnetic Resonance in Med.*, vol. 8, pp. 16–24, 1988.

[13] M. R. Mitchell, T. E. Conturo, T. J. Gruber, and J. P. Jones, "Two computer models for selection of optimal magnetic resonance imaging (MRI) pulse sequence timing," *Investigative Radiology*, pp. 349–360, Sept./Oct. 1984.

[14] L. E. Quint, et. al., "In vivo and in vitro MR imaging of renal tumors: Histopathologic correction and pulse sequence optimization," *Radiology*, vol. 169, pp. 359–362, 1988.

[15] H. Iwaoka, T. Hirata, and H. Matsuura, "Optimal pulse sequences for magnetic resonance imaging-computing accurate T1, T2, and proton density images," *IEEE Trans. Med. Imag.*, vol. 6, no. 4, pp. 360–369, Dec. 1987.

[16] G. Bielke, "A method for optimization of pulse sequence in NMR imaging," *Med. Progress through Technol.*, vol. 10, pp. 171–176, 1984.

[17] J. N. Lee and S. J. Riederer, "Optimum acquisition times of two spin echoes for MR image synthesis," *Magn. Reson. Med.*, vol. 3, pp. 634–638, 1986.

[18] E. R. McVeigh, M. J. Bronskill, and R. M. Henkelman, "Optimization of MR protocols: A statistical decision analysis approach," *Magn. Res. Med.* vol. 6, pp. 314–333, 1988.

[19] R. E. Hendrick. "Image contrast and noise," *Magnetic Resonance Imaging*, First ed., Stark and Bradly, Eds., St. Louis, MO: Mosby-Year Book, Inc., pp. 66–83, 1988.

[20] F. W. Wehrli, J. R. MacFall et al., "Mechanism of contrast in NMR imaging," *J. Comput. Assist. Tomogr.*, vol. 8, no. 3, pp. 369–380, 1984.

[21] F. W. Wehrli, R. K. Breger et al., "Quantification of contrast in clinical MR brain imaging at high magnetic field," *Inves. Radiology*, vol. 20, no. 4, pp. 360–369, 1985.

[22] B. C. Eaves and R. Saigal, "Homotopies for computing fixed points in unbounded regions," *Math. Programming*, vol. 3, pp. 225–237, 1972.

APPENDIX R2

H. Soltanian-Zadeh, A.E. Yagle, J.P. Windham, and D.O. Hearshen, "Optimization of MRI Protocols and Pulse Sequence Parameters for Eigenimage Filtering," IEEE 1992 Medical Imaging Conference, Orlando, FL, Oct. 25-31, 1992, pp. 1325-27.

This is the conference paper version of Appendix R1.

Optimization of MRI Protocols and Pulse Sequence Parameters for Eigenimage Filtering

Hamid Soltanian-Zadeh[1,2], *Andrew E. Yagle*[1], *Joe P. Windham*[2], *and David O. Hay*
[1]The University of Michigan, Ann Arbor, MI 48109, and [2]Henry Ford Hospital, Detroit, MI

Abstract

The eigenimage filter generates a composite image in which a desired feature is segmented. The signal-to-noise ratio (SNR) of the eigenimage is directly proportional to the dissimilarity between the desired and interfering features. Since image gray levels are analytical functions of MRI parameters, it is possible to maximize this dissimilarity by optimizing these parameters. For optimization, we consider four MRI pulse sequences: multiple spin-echo (MSE); spin-echo (SE); inversion recovery (IR); and gradient-echo (GE). We use the mathematical expressions for MRI signals along with intrinsic tissue parameters to express the objective function (NSNR of the eigenimage) in terms of MRI parameters. The objective function along with a set of diagnostic or instrumental constraints define a multi-dimensional non-linear constrained optimization problem, which we solve by the fixed point approach. The optimization technique is demonstrated through its application to phantom and brain images. We found that the optimal pulse sequence parameters for a sequence of four MSE and one IR images almost doubles the normalized SNR (NSNR) of the brain eigenimages, as compared to the conventional brain protocol.

I. INTRODUCTION

Optimization of MRI protocols and pulse sequence parameters has been an area of particular interest during the last few years [1]-[7]. In deriving optimal MRI protocols and pulse sequence parameters several figures-of-merit (FOM) (or objective functions (OF)) have been used. They include: (*i*) image SNR [1]; (*ii*) image contrast-to-noise ratio (CNR) [2] (sometimes normalized to the square root of the imaging time [3]); (*iii*) signal gradient (parameter sensitivity) [4]; (*iv*) accuracy of the calculated tissue parameters [5]; (*v*) noise of the synthesized image [6]; and (*vi*) diagnostic test [7].

Our problem is different from previous problems in that our FOM (OF) is not the SNR of a single image but the *normalized* SNR (NSNR) of a *composite* image (eigenimage), generated by a linear combination of several images (an MRI scene sequence) which are acquired using multiple pulse sequences. It is similar to the optimization problems (*iv*), (*v*), and (*vi*) above, in that for all of these problems multiple pulse sequences are used and hence there are several parameters to be optimized.

The eigenimage filter is the optimal transformation for MRI scene segmentation while correcting for partial volume averaging effects [8]. For the case of well-defined signature vectors, we have shown that the SNR of the eigenimage is directly proportional to [8]:

1. The reciprocal of the noise strength in the original images;

2. The dissimilarity of the desired feature from the interfering features.

We have considered methods of improving item 1 elsewhere [9], here we consider item 2. Since the image gray levels are analytical functions of MRI protocols and pulse sequence parameters, we are able to maximize the dissimilarity between the desired and interfering features by optimizing MRI protocols and pulse sequence parameters. This results in an increased SNR per unit acquisition time.

In Section II, we describe problem formulation and the approach to find a solution. In Section III, we present theoretical and experimental results for a quality control (QC) phantom and the human brain. In Section IV, we give a summary and conclusions.

II. METHODS

A. *Problem Formulation*

For optimization, we consider MRI scene sequences generated using four basic MRI protocols and pulse sequences: (*i*) multiple spin-echo (MSE); (*ii*) spin-echo (SE); (*iii*) inversion recovery (IR); and (*iv*) gradient-echo (GE). For a fixed number of images in the sequence, acquired by a certain combination of these MRI protocols, we use: (*i*) the mathematical expression of the MRI signal from a tissue (an element of a signature vector) [10]; and (*ii*) intrinsic tissue parameters (N(H), T1, and T2) [11] to express the

0-7803-0883-2/93/$3.00 ©1993 IEEE

objective function f (NSNR of the eigenimage) in terms of the MRI pulse sequence parameters (TE, TI, TR, and flip angle).

The objective function, along with a set of diagnostic or instrumental constraints on pulse sequence parameters, define a multi-dimensional non-linear constrained optimization problem. Assuming the constraints are consistent, there will be a domain of pulse sequences and pulse sequence parameters from which the optimal procedure is extracted.

The formulated optimization problem is therefore

$$\begin{aligned} &\text{Maximize} \quad f(\vec{x}) \\ &\text{subject to} \quad \begin{cases} lb_i \le x_i \le ub_i, & i = 1, \cdots, p \\ g_j(\vec{x}) \le 0, & j = 1, \cdots, m \end{cases} \end{aligned} \tag{1}$$

where f is the NSNR of the eigenimage, $\vec{x}$ contains pulse sequence parameters, $lb_i \le x_i \le ub_i$, $i = 1, \cdots, p$ implement lower and upper bounds (x_i is a varying parameter, p is the number of varying parameters, and lb_i and ub_i are lower and upper bounds, respectively.), and $g_j(\vec{x}) \le 0$, $j = 1, \cdots, m$ are linear constraints (m is the number of required relationships among the parameters).

B. *Solution*

We translate the formulated optimization problem in (1) into a fixed point problem. We then solve the fixed point problem using the method of Eaves and Saigal [12], which is the only available fixed point algorithm. It is a numerical method for solving a multi-dimensional system of equations in the form $f(\vec{x}) = \vec{x}$ defined from $\mathcal{R}^n$ to $\mathcal{R}^n$. This algorithm uses a methodology based on triangulating the space (as in finite element methods) and following a path of solutions to a minimum. This method is ideally suited for non-differentiable optimization, and thus can handle inequality and equality constraints without explicitly considering the active sets. For other methods, this can be a problem [12]. It is robust in passing the local minima and giving the global minimum, except when there exists a large peak between the local and global minima. Its rate of convergence is quadratic and is thus fast.

III. RESULTS

The optimization technique was applied to the QC phantom and the human brain. For each combination of MRI pulse sequences, the optimal parameters were found and the NSNRs of the resulting eigenimages were mathematically predicted. These results are summarized in Tables 1 and 2. It is seen that a sequence of four MSE and a GE, with optimal pulse sequence parameters, is expected to generate phantom eigenimages with largest NSNRs. Similarly, a sequence of four MSE and an IR, with optimal pulse sequence parameters, is expected to generate brain eigenimage with largest NSNRs.

Table 1: NSNRs of the QC phantom eigenimages, considering optimum parameters for three combinations of MRI pulse sequences: (a) 4 MSE (TE/TR = 31.45/498.7 msec) and 1 SE (TE/TR = 10.0/1500 msec); (b) 4 MSE (TE/TR = 22.13/500.1 msec) and 1 IR (TE/TI/TR = 20.0/776.7/2122 msec); (c) 4 MSE (TE/TR = 35.33/500 msec) and 1 GE (TE/TR = 5.0/500 msec, $\alpha = 39.38$ deg).

		$NSNR_A$	$NSNR_B$	$NSNR_C$
a	Mathematical	9.89	12.26	10.06
	Experimental	13.83	9.34	9.03
b	Mathematical	7.58	9.12	6.80
	Experimental	6.13	9.28	8.10
c	Mathematical	14.04	13.89	14.28
	Experimental	20.66	11.94	9.60

Table 2: NSNRs of the brain eigenimages, considering optimum parameters for three combinations of MRI pulse sequences: (a) 4 MSE (TE/TR =52.55/2499 msec) and 1 SE (TE/TR = 10.0/500 msec); (b) 4 MSE (TE/TR = 19.00/1500 msec) and 1 IR (TE/TI/TR = 12.0/519.0/2000 msec); (c) 4 MSE (TE/TR = 53.51/2467 msec) and 1 GE (TE/TR = 5.0/500 msec, $\alpha = 111.95$ deg).

		$NSNR_W$	$NSNR_G$	$NSNR_C$
a	Mathematical	5.78	4.59	20.55
b	Mathematical	11.28	11.77	14.39
	Experimental	10.21	12.57	13.24
c	Mathematical	5.94	4.76	21.00

Several experiments were performed to evaluate the mathematical predictions. The QC phantom was imaged using each set of pulse sequence parameters given in Table 1. Eight human volunteers were imaged using both the conventional and the optimal pulse sequence parameters for a sequence of four MSE and an IR given in Table 2. For each experiment, three eigenimages were generated and their NSNRs were estimated. Original and eigenimages of the QC phantom and two human brains were shown in the presentation, we omit them here due to the page limitation. Tables 1 and 2 compare the mathematical and experimental NSNRs of the resulting eigenimages. Table 3 compares the resulting NSNRs of eight volunteers' eigenimages using conventional MRI pulse sequence parameters for brain studies with those using the optimal MRI pulse sequence parameters. The ratio of the NSNR of the eigenimage generated using optimal parameters to that of conventional parameters illustrates the improvement attained as a result of optimization. This ratio is 1.54 ± 0.36 for white matter, 2.75 ± 1.30 for gray matter, and 1.19 ± 0.31 for cerebrospinal fluid (CSF).

Table 3: Experimental NSNRs of the brain eigenimages and the improvement ratio (ratio), considering conventional (con) MRI parameters (a MSE with TE/TR = 25/2500 msec and a SE with TE/TR = 20/500 msec) and the optimal (opt) MRI parameters (a MSE with TE/TR = 19/1500 msec and an IR with TE/TI/TR = 12/519/2000 msec).

		$NSNR_W$	$NSNR_G$	$NSNR_C$
mean values (8 volunteers)	con	6.86	5.13	11.77
	opt	10.21	12.57	13.24
	ratio	1.54	2.75	1.19
standard deviations (8 volunteers)	con	1.32	1.70	3.14
	opt	2.06	2.97	2.32
	ratio	0.36	1.03	0.31

IV. SUMMARY

We optimized MRI protocols and pulse sequence parameters for the eigenimage filtering. We formulated the maximization of the NSNR of the eigenimage as a multidimensional non-linear constrained optimization problem, which we solved by the fixed point approach.

We found the mathematical predictions for the optimal MRI parameters for the QC phantom and the human brain. We then performed several experiments to evaluate the mathematical predictions. These experiments confirmed that an improvement can always be obtained by using the optimal parameters. We, however, observed some differences between the mathematical and experimental results. We attribute these differences to the model inaccuracies and the error in estimating NSNRs and tissue parameters as well as the person-to-person variation of the tissue parameters.

The final outcome of the investigation was that on average the optimal pulse sequence almost doubled the NSNR of the brain eigenimages, as compared to the conventional brain protocol. This indicates that using the optimal protocol and pulse sequence parameters can reduce the imaging time for the eigenimage filtering of brain studies by 75%.

V. ACKNOWLEDGMENT

The authors would like to thank Professor Romesh Saigal from the University of Michigan for suggesting the fixed point approach and providing a copy his fixed point program. They also would like to thank Donald Peck, Steve Ramsey, Lucie Bower, and Linda LeBlanc from the Medical Image Analysis Laboratory of Henry Ford Hospital for their help with programming, data collection, and analysis.

VI. REFERENCES

[1] R.E. Hendrick, F.D. Newman, and W.R. Hendee, " MR Imaging Technology: Maximizing the Signal-to-Noise Ratio from a Single Tissue," *Radiology*, vol. 13, No. 3, pp. 749-752, 1985.

[2] R. M. Henkelman, et. al., "Optimal Pulse Sequence for Imaging Hepatic Metastases," *Radiology* , vol. 161, pp. 727-734, 1986.

[3] W. Dreher, and P. Bornert "Pulse Sequence and Parameter Choice in NMR Imaging as a Problem of Constrained Multidimensional Nonlinear Optimization," *Mag. Res. Med.*, vol. 8, pp. 16-24, 1988.

[4] L. E. Quint, et. al., "In Vivo and in Vitro MR Imaging of Renal Tumors: Histopathologic Correction and Pulse Sequence Optimization," *Radiology* , vol. 169, pp. 359-362, 1988.

[5] H. Iwaoka, T. Hirata, and H. Matsuura, "Optimal Pulse Sequences for Magnetic Resonance Imaging – Computing Accurate T1, T2, and Proton Density Images," *IEEE Trans. Med. Imag.*, vol. MI-6, No. 4, pp. 360-369, Dec. 1987.

[6] J.N. Lee and S.J. Riederer, "Optimum Acquisition Times of two Spin Echoes for MR Image Synthesis," *Magn. Reson. Med.*, vol. 3, pp. 634-638, 1986.

[7] E.R. McVeigh, M.J. Bronskill, and R.M. Henkelman, "Optimization of MR Protocols: A Statistical Decision Analysis Approach," *Magn. Res. Med.* vol. 6, pp. 314-333, 1988.

[8] H. Soltanian-Zadeh, J.P. Windham, D.J. Peck, and A.E. Yagle,"A Comparative Analysis of Several Transformations for Enhancement and Segmentation of Magnetic Resonance Image Scene Sequences," *IEEE Trans. Med. Imag.*, vol. 11, No. 3, pp. 2-18, Sept. 1992.

[9] H. Soltanian-Zadeh, J.P. Windham, and A.E. Yagle: "Magnetic Resonance Image Restoration Using a New Multi-Dimensional Non-Linear Edge-Preserving Filter." *Submitted to the IEEE Trans. Imag. Proc.*, May 1992.

[10] R.E. Hendrick. Image Contrast and Noise, chapter 5 in *Magnetic Resonance Imaging*, First Edition, Editors: Stark and Bradly, Mosby-Year Book, Inc., St. Louis, MO, pages 66-83, 1988.

[11] F.W. Wehrli, R.K. Breger, et al., " Quantification of Contrast in Clinical MR Brain Imaging at High Magnetic Field," *Inves. Radiology*, vol. 20, No. 4, pp. 360-369, 1985.

[12] B.C. Eaves and R. Saigal, "Homotopies for Computing Fixed Points in Unbounded Regions," *Math. Programming*, vol. 3, pp. 225–237, 1972.

APPENDIX S

H. Soltanian-Zadeh, J.P. Windham and A.E. Yagle, "A Multidimensional Non-Linear Edge-Preserving Filter for Magnetic Resonance Image Restoration," to appear in IEEE Trans. Image Proc.

Although the edge-preserving filter with locally-varying properties was designed specifically for MRI, it may have applications elsewhere.

A Multidimensional Nonlinear Edge-Preserving Filter for Magnetic Resonance Image Restoration

Hamid Soltanian-Zadeh, *Member, IEEE*, Joe P. Windham, *Member, IEEE*, and Andrew E. Yagle, *Member, IEEE*

Abstract—**This paper presents a multidimensional nonlinear edge-preserving filter for restoration and enhancement of magnetic resonance images (MRI). The filter uses both interframe (parametric or temporal) and intraframe (spatial) information to filter the additive noise from an MRI scene sequence. It combines the approximate maximum likelihood (equivalently, least squares) estimate of the interframe pixels, using MRI signal models, with a trimmed spatial smoothing algorithm, using a Euclidean distance discriminator to preserve partial volume and edge information. (Partial volume information is generated from voxels containing a mixture of different tissues.) Since the filter's structure is parallel, its implementation on a parallel processing computer is straightforward. Details of the filter implementation for a sequence of four multiple spin-echo images is explained, and the effects of filter parameters (neighborhood size and threshold value) on the computation time and performance of the filter is discussed. The filter is applied to MRI simulation and brain studies, serving as a preprocessing procedure for the eigenimage filter. (The eigenimage filter generates a composite image in which a feature of interest is segmented from the surrounding interfering features.) It outperforms conventional pre and post-processing filters, including spatial smoothing, low-pass filtering with a Gaussian kernel, median filtering, and combined vector median with average filtering.**

I. Nomenclature

For the mathematical developments, we use an n-dimensional vector space $(\mathcal{R}^n, \mathcal{R})$, where n is the number of images in the MRI scene sequence. For example, when dealing with an MRI scene sequence consisting of a T1-weighted and four T2-weighted spin-echo images, we use a 5-D vector space. Using the vector space concept, the following representations are introduced. The MRI scene sequence is represented by *pixel vectors*. A pixel vector $\vec{P}_{jk} = [P_{jk1}\ P_{jk2}\ \cdots\ P_{jkn}]^T$ is a vector whose elements are the corresponding gray levels of the (j,k)th pixels in the MR images (see Fig. 1). The image size determines the number of these pixel vectors, e.g., for 256 × 256 images, there are 2^{16} pixel vectors. The MRI characteristics of tissue types are represented by *signature vectors*. For image analysis, one is normally interested in clearly visualizing one of the tissue types (which is referred to as the desired feature), whereas other tissue types (which are referred to as the undesired or interfering features) interfere with its visualization. A *desired signature vector* $\vec{d} = [d_1\ d_2 \cdots d_n]^T$ is defined as a vector whose ith element is the average gray level of the desired feature in the ith image. *Undesired (interfering) signature vectors* $\vec{u}_i = [u_{1i}\, u_{2i} \cdots u_{ni}]^T, 1 \le i \le m$ are similarly defined for the interfering features. Finally, vectors $\vec{P}^d_{jk}$ and $\vec{P}^u_{lm}$ are pixel vectors from the desired feature at location (j,k) and the undesired feature at location (l,m), respectively. These notations, as well as those defined elsewhere in the paper, are summarized in a list below the following list of abbreviations.

1) List of Abbreviations

AVG	average
CNR	contrast-to-noise ratio
SNR	signal-to-noise ratio
CSF	cerebrospinal fluid
DROI	desired feature ROI
UROI	undesired feature ROI
EPV	estimated partial volume
OPV	original partial volume
LS	least squares
MLE	maximum likelihood estimate
MR	magnetic resonance
MRI	magnetic resonance imaging/images
NS	neighborhood size
P_D	probability of detection
P_F	probability of false alarm
ROI	region of interest

2) List of Notations

CNR_i	CNR between the desired and the ith interfering feature
$\vec{d}$	desired feature signature vector
$\vec{e}$	weighting vector for the eigenimage filter
$E[\cdot]$	expected value operator
$\hat{E}[\cdot]$	expected value estimator (sample mean)
EI_{jk}	gray level of the (j,k)th pixel in the eigenimage
m	number of interfering features in the scene
n	number of images in the MRI scene sequence
N	number of pixels in the DROI
P_{jk}	gray level of the (j,k)th pixel in an image
P_{jki}	gray level of the (j,k)th pixel in the ith image
$\vec{P}_{jk}$	pixel vector, i.e., an n-dimensional vector whose ith element is P_{jki}

Manuscript received May 24, 1992; revised October 13, 1993. This work was supported by NIH under Grant 1R29-CA61263. Initial results of this work were presented at the 10th Annual Meeting of the Society of Magnetic Resonance in Medicine (SMRM). The associate editor coordinating the review of this paper and approving it for publication was Prof. Roland T. Chin.

H. Soltanian-Zadeh and A. E. Yagle are with the Department of Electrical Engineering and Computer Science, the University of Michigan, Ann Arbor, MI 48109-2122 USA.

J. P. Windham is with the Department of Diagnostic Radiology and Medical Imaging, Henry Ford Hospital, Detroit, MI 48202 USA.

IEEE Log Number 9407595.

1057–7149/95$04.00 © 1995 IEEE

P^d_{jk}	gray level of the (j,k)th pixel in the DROI of an image
$\vec{P}^d_{jk}$	pixel vector in the DROI
P^u_{jk}	gray level of the (j,k)th pixel in an UROI of an image
$P^{u_i}_{jk}$	gray level of the (j,k)th pixel in the ith UROI of an image
$\vec{P}^{u_i}_{jk}$	pixel vector in the ith UROI
σ	standard deviation of white noise
SNR_d	SNR of the desired feature
S_i	MRI signal from the ith material
$\vec{u}$	undesired feature signature vector
$\vec{u}_i$	undesired feature signature vector
$\mathrm{Var}(\cdot)$	variance operator
$\hat{\mathrm{Var}}(\cdot)$	variance estimator (sample variance)
V_l	partial volume of the lth material in a voxel
V_{ljk}	partial volume of the lth material in the (j,k)th voxel
V	total volume of a voxel
$\vec{w}$	weighting vector for a linear filter
w_{jk}	zero-mean white noise at the (j,k)th pixel of an image
w_{jki}	zero-mean white noise at the (j,k)th pixel of the ith image

II. Introduction

Additive noise in magnetic resonance imaging (MRI) limits correct identification and quantitative measurements of normal and pathological tissues. This is a problem for human viewers as well as computer vision and automatic analysis methods such as image segmentation and analysis using a modified matched filter [1] and the eigenimage filter [2]–[4]. The eigenimage filter will be briefly described in Section III-E.

In MRI clinical studies, a sequence of images of the same anatomical site is usually acquired, which we refer to as an *MRI scene sequence*. (This is similar to multispectral images acquired in the field of remote sensing.) The noise in an MRI scene sequence is characterized by an additive zero-mean white Gaussian noise field that is uncorrelated between different frames [5]–[7].

Averaging several acquisitions or free induction decay signals, which are used to reconstruct magnetic resonance (MR) images, is the conventional method for reducing the additive noise. This method has the following practical difficulties:

1) It increases the imaging time and cost.
2) It limits the patient throughput.
3) It requires image registration to compensate for the patient movements.

An alternative approach is restoration of the acquired MR images, which is performed off-line without having the aforementioned difficulties. However, conventional image restoration filters found in the image processing literature, e.g., Wiener, median, and low-pass filters, are not specifically designed for MRI. As such, they neither use all of the available information in MRI nor consider the MRI specific requirements, e.g., preserving partial volume and edge information. The concept of partial volume and its basis will be explained in Section III-C.

Recent developments in nonlinear filters for computer vision, e.g., scale space and edge detection using anisotropic diffusion [8] and adaptive smoothing [9], are indications of a need for new nonlinear methods to preserve edges while suppressing the additive noise. Both of these filters use a 3×3 neighborhood and a nonlinear function of the signal gradient to preserve edges. They are most appropriate for processing a single image with sharp transitions between different regions.

In an MRI scene sequence, there are usually partial volume pixels that generate a smooth transition between different regions. In addition, there are several images that may be processed simultaneously. Therefore, a new multidimensional nonlinear edge-preserving filter specifically designed for MRI seems necessary. We have developed such a filter that uses both intraframe (spatial) and interframe (parametric or temporal) information to filter the noise while preserving edge and partial volume information.

The new filter uses MRI signal models to implement an approximate maximum likelihood or least squares estimate of each pixel gray level from the gray levels for the same location in all of the images in sequence; this corresponds to using interframe information. It also employs a trimmed mean spatial smoothing algorithm[1] that uses a Euclidean distance discriminator to preserve partial volume and edge information; this corresponds to using intraframe information.

Trimming data by the new filter is reminiscent of that previously used in generalized order statistic filters [10]–[15] and in combination with segmentation algorithms [16]. Nonlinearity and adaptivity of the new filter is similar to that used for anisotropic diffusion [8] and adaptive smoothing [9]. However, there are some differences between the new filter and previous methods. The most important differences are the following:

1) It uses a Euclidean distance discriminator (i.e., the optimal solution to a binary detection problem [17]).
2) It puts no restriction on the number of trimmed data points.
3) It uses a step function as its nonlinearity.
4) It finds the edge of the step function by calculating probabilities of detection and false alarm.

In Section III, we review the MRI signal model for multiple spin-echo images and describe interframe and intraframe information, partial volume averaging, and signal-to-noise ratio (SNR) and contrast-to-noise ratio (CNR) as defined and estimated in the MRI literature. We also briefly review the eigenimage filter. In Section IV, we explain the relationship between the new filter and the generalized order statistic filters and adaptive smoothing methods. In Section V, we explain details of the new filter including the implementation of the approximate maximum likelihood and least squares estimates and calculation of the probabilities of detection and false alarm. In Section VI, we apply the new filter to preprocess multiple spin-echo images of a simulation and

[1] Trimmed mean spatial smoothing refers to the idea of ignoring some data points in a neighborhood around each pixel as being irrelevant and using the sample average of the remaining data points as an estimate for the point in the center of the neighborhood.

a brain study before the eigenimage filtering. We illustrate the effects of neighborhood size and threshold value on the CNR and appearance of the resulting eigenimage using simulated and acquired brain images. We then compare the performance of the new filter to that of several conventional pre- and post-processing methods, including spatial smoothing, low-pass filtering with a Gaussian kernel, median filtering, and combined vector median with average filtering. Finally, we illustrate the preservation of edge and partial volume information by the new filter using a 1-D signal extracted from the simulation and volume calculations of the central region (simulated white matter) in the simulation and the egg white and egg yolk in an egg phantom. Conclusions and comments are given in Section VII.

III. Background

A. Multiple Spin-Echo Model

To acquire MR images, a sequence of radio-frequency (RF) pulses with certain shapes, energy levels, and timing intervals are applied while the object being imaged is immersed in a static magnetic field. The resulting average image gray level, corresponding to a specific tissue type, is a function of both the intrinsic tissue parameters and the parameters of the RF pulse sequence. One of the most commonly used pulse sequences is the *multiple spin-echo* (which is also referred to as CPMG) [7], which can generate several images (clinically utilized to generate two or four images) in a single acquisition setup. An MRI scene sequence, which is defined by n multiple spin-echo images of a slice with k tissue types in the scene, can be thought of as a set of n-dimensional random vectors, each pertaining to one of k uncorrelated Gaussian distributions. The MRI pulse sequence and tissue parameters determine the standard deviation of each distribution and the relationship between their mean vectors.

Here, we review the theoretical model for the image gray levels generated by a multiple spin-echo sequence with n echoes in terms of tissue and pulse sequence parameters. The MRI signal S_i (which is the deterministic portion of the image gray level) arising from a region with tissue specific parameters $N(H)$ (proton density) and relaxation times $T1$ and $T2$ in the ith image is given by

$$S_i = N(H)\left[1-(-1)^n\left[2\sum_{l=1}^{n}(-1)^l e^{(2l-1)TE-2TR/2T1} + e^{-TR/T1}\right]\right]e^{-iTE/T2}. \quad (1)$$

Equation (1) illustrates that for a particular tissue, the signal is a decaying exponential function of the image number i. With the additive white noise, the intensity of the (j,k)th pixel in the ith image P_{jki} can be represented as

$$P_{jki} = M_{jk}e^{-iTE/T2_{jk}} + w_{jki} \quad (2)$$

where M_{jk} is a function of $T1$ and $N(H)$ for the tissue at position (j,k) as well as TE and TR of the pulse sequence as given in (1), and w_{jki} represents white Gaussian noise.

B. Inter and Intraframe Information

The model in (2) shows the relationship between corresponding pixels from different images, which we refer to as interframe (parametric[2] or temporal) information. This model provides a means for obtaining a least-squares (LS) or maximum likelihood estimate (MLE) of the pixel intensities. The estimation details will be explained in Section V-A.

Intraframe (spatial) information refers to the relationship between pixels from a particular tissue, which is provided by anatomical structures visualized in the image. This information exists because each tissue type has several *connected* pixels in the image. Our approach to utilize this information will be explained in Section V.

C. Partial Volume Averaging

In this section and throughout the rest of the paper, whenever it is possible, we present definitions and explanations using a single image. This is done for the purpose of notation simplicity. The concepts can be easily extended to multiple images.

Here, we review the theoretical model for the MRI signal generated from voxels containing a mixture of multiple tissues in terms of the signals for each of the tissues. The MRI signal S from a voxel containing m different materials is given by [19]

$$S = \sum_{l=1}^{m}\left(\frac{V_l}{V}\right)S_l \quad (3)$$

where

- V_l volume of the lth material within the voxel
- V total volume of the voxel
- S_l signal from the lth material.

The gray level P_{jk} of the (j,k)th pixel (corresponding to the (j,k)th voxel) in an MR image is given by

$$P_{jk} = E[P_{jk}] + w_{jk} = \sum_{l=1}^{m}\left(\frac{V_{ljk}}{V}\right)S_l + w_{jk} \quad (4)$$

where V_{ljk} is the partial volume of the lth material in the (j,k)th voxel, and w_{jk} represents statistical noise that is again assumed to be an additive zero-mean white Gaussian noise field with standard deviation σ. Note that $E[P_{jk}]$ is deterministic but unknown, whereas the noise w_{jk} is stochastic; therefore, the pixel gray level P_{jk} is the sum of a deterministic value (to be estimated) and noise. We use the notation $E[P_{jk}]$ to denote the deterministic value of the pixel gray level.

Preserving partial volume averaging information means that the deterministic portion of the image gray levels is maintained, whereas the stochastic portion is suppressed by the restoration filter.

D. Signal-to-Noise and Contrast-to-Noise Ratios

For medical image interpretation, one is normally interested in clearly visualizing a specific feature (tissue type), which is

[2] This is sometimes misleadingly called spectral, which is taken from remote sensing terminologies.

referred to as the desired feature. To assess the image quality, one thus may consider the SNR of this desired feature (SNR_d), which, in the MRI literature, is defined as

$$SNR_d = \frac{E[P^d_{jk}]}{\sqrt{\mathrm{Var}(P^d_{jk})}} \tag{5}$$

where $E[P^d_{jk}]$ is the expected value of the (j,k)th pixel gray level in a region of interest drawn over the desired feature, which is called the desired ROI (DROI), and $\sqrt{\mathrm{Var}(P^d_{jk})}$ is the standard deviation of the noise in the DROI. Since statistical noise in MRI is ergodic and uncorrelated with the signal [5], [6], $E[P^d_{jk}]$ and $\mathrm{Var}(P^d_{jk})$ in a homogeneous region of N pixels can be estimated using the sample mean and variance [20]

$$\hat{E}[P^d_{jk}] = \frac{1}{N}\sum_{j,k} P^d_{jk}, \tag{6}$$

$$\hat{\mathrm{Var}}(P^d_{jk}) = \frac{1}{N-1}\sum_{j,k}\left(P^d_{jk} - \frac{1}{N}\sum_{j,k} P^d_{jk}\right)^2. \tag{7}$$

The estimated values $\hat{E}[P^d_{jk}]$ and $\hat{\mathrm{Var}}(P^d_{jk})$ are then inserted into (5).

To derive an analytic expression for the SNR_d of a composite image, the following common assumptions [1], [4], [21]–[23] are used:

1) Statistical noise in MRI is modeled as a Gaussian distributed zero-mean white noise field with standard deviation σ.
2) Signature vectors are *a priori* known fairly well.

Then, the standard formula for noise propagation [24] shows that SNR_d of a linearly transformed image (e.g., the eigenimage) with the weighting vector $\vec{w}$ is simplified to [4]

$$SNR_d = \frac{\vec{w}\cdot\vec{d}}{\sigma(\vec{w}\cdot\vec{w})^{1/2}}. \tag{8}$$

An estimate of σ^2 is found by (7) using image gray levels in the DROI or those in a background ROI. When using magnitude reconstructed images, the standard deviation found from the background ROI needs to be divided by 0.655 to yield the noise standard deviation [25].

The CNR between the desired feature (tissue) and the ith interfering feature (background or another tissue type), which is denoted by CNR_i, is usually more important than the SNR_d for medical image interpretation. It quantifies the distinguishability of the desired feature from the interfering features. The CNR_i is defined as [21]

$$CNR_i = \frac{E[P^d_{jk}] - E[P^{u_i}_{jk}]}{\sqrt{\dfrac{\mathrm{Var}(P^d_{jk}) + \mathrm{Var}(P^{u_i}_{jk})}{2}}} \tag{9}$$

where $P^{u_i}_{jk}$ is the gray level of the (j,k)th pixel in the undesired ROI (UROI), and $E[P^u_{jk}]$ and $\mathrm{Var}(P^u_{jk})$ are the mean and the variance of pixel values in the UROI, respectively.

E. Eigenimage Filter

The eigenimage filter maximizes the projection of a desired feature while minimizing the projections of the undesired (interfering) features in a composite image called an eigenimage (EI) [2]. It has been shown that it maximizes the SNR_d while correcting for partial volume averaging effects [4], [26]. Since, in the eigenimage, the desired feature appears bright, the interfering features appear dark, and the partial volumes of the desired feature are visualized. Viewing the eigenimage would be helpful for a better interpretation of the MRI scene. Fig. 3 illustrates the application of the eigenimage filter for segmenting gray matter from white matter and cerebrospinal fluid (CSF) in brain images.

Mathematically, the eigenimage is a weighted sum of the images in the sequence. A pixel in the eigenimage is therefore a linear combination of all pixels at the same location in the MR images, i.e.

$$EI_{jk} = \sum_{i=1}^{n} e_i P_{jki} = \vec{e}\cdot\vec{P}_{jk} \tag{10}$$

where EI_{jk} is the gray level of the (jk)th pixel in the eigenimage, and $\vec{e} = [e_1\, e_2 \cdots e_n]^T$ is the weighting vector to be determined.

To determine the weighting vector, the SNR_d is maximized

$$\mathbf{Max.}\left[SNR_d = \frac{\vec{e}\cdot\vec{d}}{\sigma(\vec{e}\cdot\vec{e})^{1/2}}\right] \tag{11}$$

subject to the constraint that

$$\vec{e}\cdot\vec{u}_i = 0, \qquad \text{for} \quad i = 1,\cdots,m. \tag{12}$$

The solution to the above constrained optimization problem is given by [4], [26]

$$\vec{e} = \vec{d} - \vec{d}^{\,\prime} \tag{13}$$

where $\vec{d}^{\,\prime}$ is the projection of $\vec{d}$ onto the subspace spanned by $\{\vec{u}_i, i = 1,\cdots,m\}$ and can be computed using a Gram–Schmidt orthogonalization procedure. The solution is always nonzero unless $\vec{d}$ is linearly dependent on $\{\vec{u}_i, i = 1,\cdots,m\}$, which is very unlikely to happen in practice.

IV. Relation with Similar Filters

Before presenting the new image restoration filter in Section IV, we explain the relationship of this new filter to the generalized order statistic filters and to the anisotropic diffusion and adaptive smoothing filters.

A. Generalized Order Statistic Filters [10]–[15]

For this class of filters, the data points in a neighborhood from a *single* image are ordered. A *fixed* percentage of the upper and lower ends of the ordered data is ignored as outliers, and the sample average of the remaining data points is used as an estimate for the point in the center of the

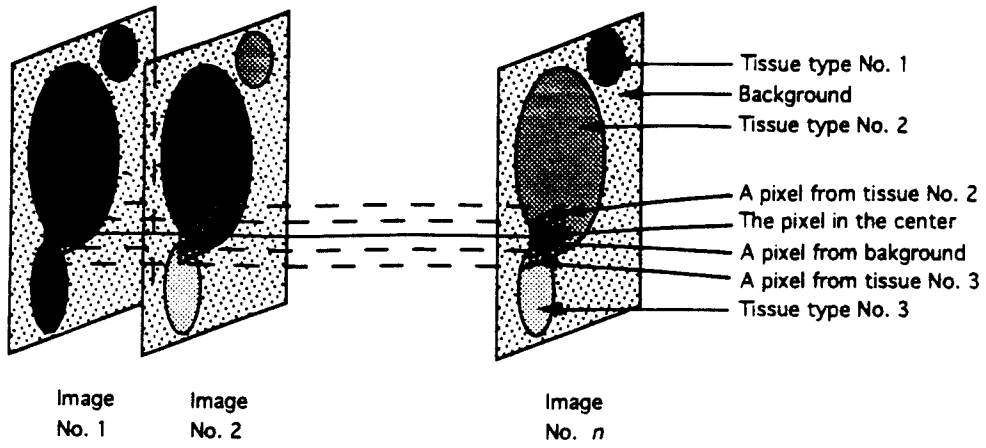


Fig. 1. Graphic illustration of an MRI scene sequence and the use of a neighborhood to access both of intraframe (spatial) and interframe (parametric or temporal) information.

neighborhood.[3] The new filter shares the idea of trimming data with this class of filters. It, however, considers a *set* of images simultaneously, rather than a *single* image. It neither orders the data points nor fixes the percentage of the data to be ignored. Instead, it uses the Euclidean distance as a similarity measure and discards any pixel vector whose Euclidean distance is larger than a specific threshold value (this is adopted from the optimal solution to a binary detection problem [17]). The threshold is selected based on the probabilities of detection and false alarm, as explained in Sections V-B and C. A detailed analysis of the threshold selection is a unique feature of this paper.

B. Anisotropic Diffusion and Adaptive Smoothing [8], [9]

These filters consider a *single* image and a 3 × 3 neighborhood. They use a nonlinear function of the magnitude of the gradient vector at each point in the neighborhood as the weighting factor for smoothing. The new filter shares the idea of adapting the weighting factor to the characteristics of the signal at each point of the neighborhood with these filters. It, however, considers a *set* of images simultaneously, and uses a *simple* nonlinearity (a step function) along with the Euclidean distance as a measure of similarity. It should be noted that there is little advantage in using a *smooth* function in determining the weighting factors for nonlinear filtering, as opposed to that for linear filtering that avoids frequency domain sidelobes. The main advantage of nonlinear filtering is that it discriminates between high-frequency components of signal and noise. For nonlinear filtering, the choice of a step function has two advantages: It lessens the computational load and makes it simple to optimize the nonlinearity (i.e., its threshold) based on the probabilities of detection and false alarm. Using the Euclidean distance discriminator has a major advantage of making it possible to use either a small neighborhood, e.g., a 3 × 3, or a larger one, e.g., a 9 × 9. This is impossible with the gradient vector since it makes no distinction between flat (homogeneous) regions of two different objects that may be present in a large neighborhood.

[3] A 50% trimming yields the median filter.

V. Proposed New Image Restoration Filter

The new filter uses both intra-frame (spatial) and inter-frame (parametric or temporal) information to suppress the additive noise in MR images, while preserving and enhancing edge and partial volume information. Details of the proposed filter are as follows.

First, we consider a neighborhood centered on the pixel to be estimated (see Fig. 1); the size and shape of this neighborhood depends on the size and shape of the objects in the scene as well as the allowed computation time. Since there is usually no *a priori* knowledge regarding the shapes of the objects in the scene, we always use a square neighborhood. The size of this square neighborhood is mainly limited by the allowed computation time. We normally use a 9 × 9 neighborhood for which the computation time for 256 × 256 images is about 4 min on a Sun SPARCstation 2. Use of a neighborhood larger than 9 × 9 (81 pixels) would usually improve performance, but the amount of computation required was judged to make such choices infeasible for present computational power available in typical image analysis laboratories.

Second, we implement the optimal solution to a binary detection problem [17]. That is, we calculate the Euclidean distance between each pixel vector in the neighborhood and the pixel vector in the center, which is the pixel vector to be estimated. If this distance is smaller than a specific threshold value η, we consider that pixel vector in the estimate of the pixel vector in the center; if this distance is greater than η, that pixel vector is not used in the estimate. The threshold η depends on the noise standard deviation σ in the images, the contrast between adjacent ROI's, and partial volume averaging effects that are reflected in the smoothness of the edges in each image. In practice, η is chosen based on the probabilities of detection and false alarm (as explained in Sections V-B and C), which depend on all of these. There is a tradeoff between suppression of statistical noise, preservation of the average partial volume information, and enhancement and sharpening of image edges, which is manifested as the tradeoff between the probabilities of detection and false alarm.

Third, we consider two methods of computing an estimate for the contributing pixel vectors: 1) just the sample averaging

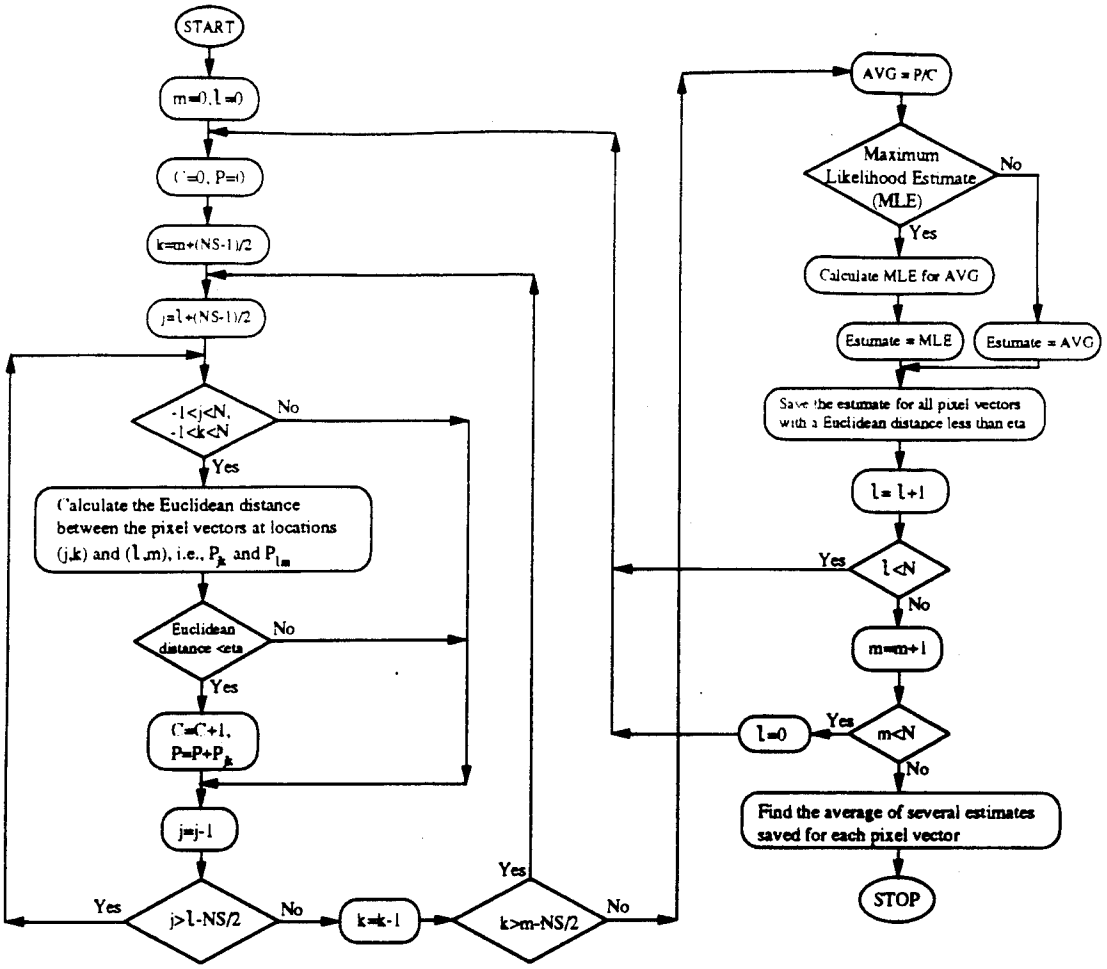


Fig. 2. Flowchart of the MLE + AVG and AVG restoration filters for an image size of $N \times N$ and a neighborhood size of $NS \times NS$.

of the contributing pixel vectors and 2) a maximum likelihood estimation (as explained in Section V-A) in addition to the sample averaging. The resulting filters are called AVG and MLE + AVG, respectively. The MLE + AVG version is expected to have a superior performance than the AVG version since it uses interframe as well as intraframe information. However, it is not applicable if the MRI signal model is inaccurate due to the specific choice of the imaging and pulse sequence protocols and parameters.

Fourth, we move to an adjacent pixel, i.e., shift the neighborhood, and repeat the procedure. Finally, we calculate the average of several estimates obtained for a particular pixel vector to find the filter output for that pixel vector. A flowchart explaining both versions of the filter is given in Fig. 2.

Note that the last averaging step uses appropriate pixels (with Euclidean distances less than η) outside a neighborhood but inside a shifted neighborhood (with shifts of at most half of the neighborhood size in each direction) in estimating the pixel vector in the center of the first neighborhood. This allows contributions from the appropriate pixels in a larger neighborhood encompassing quadruple number of pixels with the addition of a small computational load. Considering this effective neighborhood size, the upper bound on the SNR improvement factor for the AVG version of the new filter is $2NS$, using a $NS \times NS$ neighborhood.

A. Approximate Maximum Likelihood and Least Squares Estimates

Maximum likelihood and least-squares are two well-known estimation criteria. For the Gaussian pixel model in (2), the maximum likelihood estimate coincides with the least-squares estimate [27]. Computation of the optimal estimator requires solving a nonlinear system of equations, which is computationally intense. We therefore consider the following suboptimal estimator of the gray levels. We perform an approximation by factoring the signal $M_{jk}e^{-iTE/T2_{jk}}$ out of (2) and then taking its natural logarithm, yielding

$$\begin{aligned}\ln(P_{jki}) &= \ln(M_{jk}e^{-iTE/T2_{jk}}) \\ &\quad + \ln\left(1 + \frac{w_{jki}}{M_{jk}e^{-iTE/T2_{jk}}}\right) \\ &\simeq \ln(M_{jk}e^{-iTE/T2_{jk}}) + \frac{w_{jki}}{M_{jk}e^{-iTE/T2_{jk}}} \\ &= \ln(M_{jk}) - \frac{iTE}{T2_{jk}} + ws_{jki} \\ &= a_{jk} + b_{jk}i + ws_{jki}. \end{aligned} \tag{14}$$

In simplifying (14), we used the Taylor series expansion

$$\ln(1+x) = \sum_{k=1}^{\infty} \frac{(-1)^{k+1}}{k} x^k \tag{15}$$

and neglected x^2 and higher order terms. This is reasonable since usually, $M_{jk}e^{-iTE/T2_{jk}}/w_{jki} \gg 1$, and thus, $x = (M_{jk}e^{-iTE/T2_{jk}}/w_{jki}^{-1}$. Note that ws_{jki} is also zero-mean and Gaussian distributed since it is a scaled version of w_{jki}. Therefore, the maximum likelihood estimate of $\ln(P_{jki})$ is identical to a weighted least squares estimate for the line $a_{jk} + b_{jk}i$. Weights for the least squares estimate are $[M_{jk}e^{-iTE/T2_{jk}}]^{-1}$, which we approximate by P_{jki} (see (2)).

B. Probability of Detection

In the selection of the threshold η, we consider the probabilities of detection P_D and false alarm P_F.[4] The probability of detection P_D is the chance of correctly identifying a pixel vector $\vec{P}_{jk}^{\,d}$ in the neighborhood that represents the same tissue type as the pixel vector $\vec{P}_{lm}^{\,d}$ in the center. These pixel vectors are assumed to be uncorrelated and Gaussian distributed, with mean vector $\vec{d}$ and covariance matrix $\sigma^2 I$. The difference vector $\vec{D}_{jk}^{\,d} = \vec{P}_{lm}^{\,d} - \vec{P}_{jk}^{\,d}$ is therefore Gaussian distributed with mean vector $\vec{0}$ and covariance matrix $2\sigma^2 I$. The square of the Euclidean distance $ED_{jk}^2 = \|\vec{D}_{jk}^{\,d}\|^2$ between these pixel vectors has a scaled chi-squared distribution with n degrees of freedom, mean of $2n\sigma^2$, and variance of $8n\sigma^4$ [28]. Using the threshold value η for the Euclidean distance ED_{jk} yields the following probability of detection P_D:

$$P_D = \int_0^{\eta^2/2\sigma^2} f_X(x)dx = \frac{1}{2^{n/2}\Gamma\left(\frac{n}{2}\right)} \int_0^{\eta^2/2\sigma^2} x^{n/2-1}e^{-x/2}dx. \quad (16)$$

Equation (16) illustrates that P_D is directly proportional to the ratio of η to σ; increasing η or decreasing σ increases P_D. The value of σ is dictated by the MRI instrumentation and acquisition parameters that are fixed from this paper's point of view. Thus, we investigate the relationship between P_D and η. To make the results independent of the *numerical* value of σ, we find P_D for η being equal to a multiple of σ. Table I lists the corresponding threshold values for several P_D's when using a sequence of four multiple spin-echo images.

C. Probability of False Alarm

The probability of false alarm P_F is the chance of wrongly classifying a pixel vector $\vec{P}_{jk}^{\,u}$, which represents a different tissue, as one that represents the same tissue as that of the pixel vector $\vec{P}_{lm}^{\,d}$ in the center of the neighborhood. As before, these pixel vectors are assumed to be uncorrelated and Gaussian distributed with identical covariance matrices $\sigma^2 I$ but different mean vectors $\vec{u}$ and $\vec{d}$, respectively. The difference vector $\vec{D}_{jk}^{\,u} = \vec{P}_{lm}^{\,d} - \vec{P}_{jk}^{\,u}$ is then Gaussian distributed with mean vector $\vec{m} = \vec{d} - \vec{u}$ and covariance matrix $2\sigma^2 I$. A scaled version of the square of the Euclidean distance between these pixel vectors, i.e., $ED_{jk}^2/2\sigma^2 = \|\vec{D}_{jk}^{\,u}\|^2/2\sigma^2$, has a *noncentral* chi-squared distribution. Using standard techniques for deriving probability density functions (pdf's) [20], the pdf

[4] It is ideal to have a P_D equal to 1.0 and a P_F equal to 0.0.

TABLE I
THRESHOLD VALUES (η) AND PROBABILITIES OF DETECTION (P_D) AND FALSE ALARM (P_F) FOR SIMULATION (BRAIN), USING WHITE MATTER AND GRAY MATTER SIGNATURE VECTORS, AND SEVERAL PERCENTAGES OF PARTIAL VOLUMES: a = 0%; b = 25%; c = 50% (σ = 8.75)

η	P_D	P_F (a)	P_F (b)	P_F (c)
1.459σ	0.100	0.0000	0.0000	0.0035
1.816σ	0.200	0.0000	0.0002	0.0086
2.095σ	0.300	0.0000	0.0004	0.0167
2.347σ	0.400	0.0000	0.0009	0.0275
2.590σ	0.500	0.0000	0.0017	0.0440
2.844σ	0.600	0.0000	0.0031	0.0675
3.124σ	0.700	0.0001	0.0053	0.1028
3.460σ	0.800	0.0001	0.0103	0.1586
3.674σ	0.850	0.0002	0.0157	0.2038
3.944σ	0.900	0.0006	0.0259	0.2675
4.356σ	0.950	0.0017	0.0516	0.3831
4.721σ	0.975	0.0035	0.0876	0.4925
5.153σ	0.990	0.0082	0.1503	0.6212

for the square of the ith component of the difference vector divided by $\sqrt{2}\sigma$ is obtained as[5]

$$f_{Y_i}(y) = \frac{1}{2\%sqrt2\pi y}[e^{-(\sqrt{y}-m_i)^2/2} + e^{-(\sqrt{y}+m_i)^2/2}], \quad y > 0, \quad 1 \le i \le n, \quad (17)$$

where m_i is the ith component of the difference vector $\vec{m}$ divided by $\sqrt{2}\sigma$. The pdf $f_Y(y)$ for the Euclidean distance squared ED_{jk}^2 divided by $\sqrt{2}\sigma$ is found by convolving n pdf's given in (17), i.e.

$$f_Y(\cdot) = f_{Y_1}(\cdot) * f_{Y_2}(\cdot) * \cdots * f_{Y_n}(\cdot). \quad (18)$$

Finally, the probability of false alarm P_F is

$$P_F = \int_0^{\eta^2/2\sigma^2} f_Y(y)\,dy. \quad (19)$$

Equations (17) to (19) show that P_F is related to the ED_{jk}/σ in addition to η/σ. That is, both the CNR's of the MR images and the threshold value determine P_F. The CNR's are dictated by the MRI instrumentation and acquisition parameters as well as the tissues in the scene, both of which are fixed from this paper's point of view; we focus on the multiple spin-echo images of the brain. We therefore investigate the relationship between P_F and η. Again, to make the results independent of the *numerical* value of σ, we find P_F for η being equal to a multiple of σ.

Table I shows the corresponding P_F's for several threshold values using the human brain images, considering different percentages of partial volumes between white matter and gray matter. For (a), it is assumed that pixel vectors in a neighborhood correspond to voxels containing either pure white matter or pure gray matter. For (b), it is assumed that pixel vectors in a neighborhood correspond to voxels containing either pure white matter or a combination of 25% white matter and 75% gray matter. Similarly for (c), pixel

[5] Division by $\sqrt{2}\sigma$ simplifies the derivation and results in a standard known distribution, i.e., noncentral chi-squared.

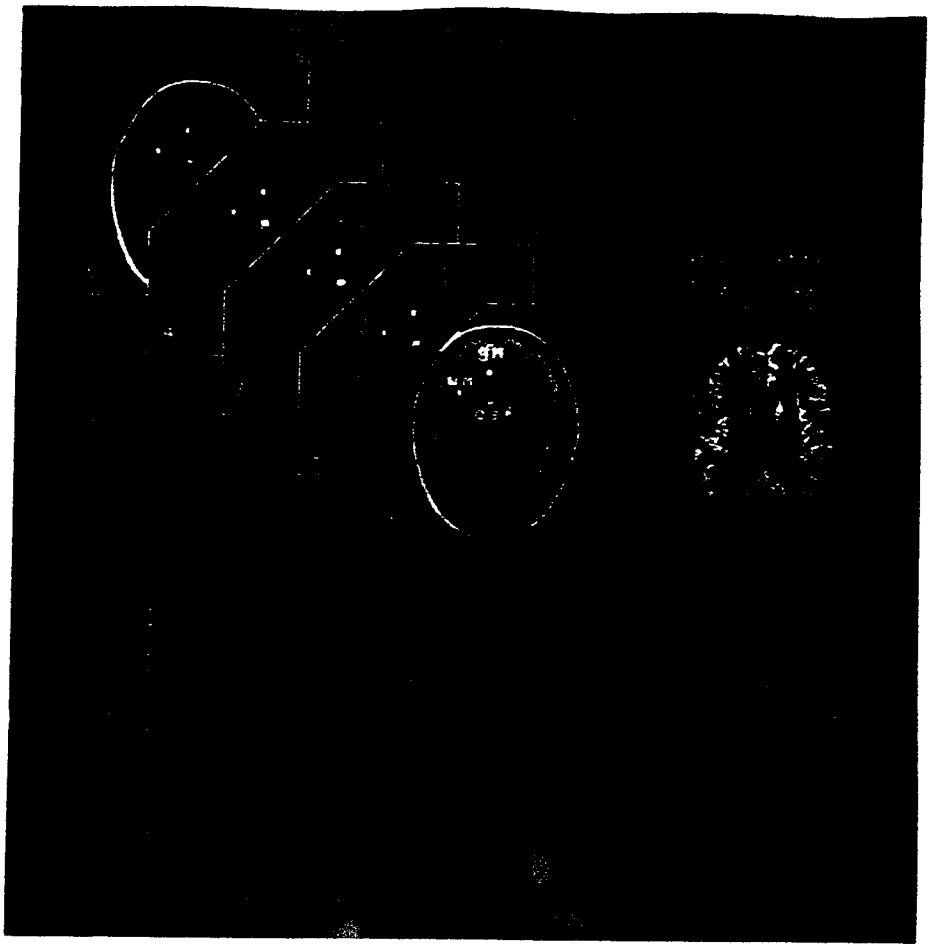


Fig. 3. Illustration of the eigenimage filtering using four multiple spin-echo T2-weighted images (TE/TR = 25–100/2500 ms) and an inversion recovery image (TE/TI/TR = 20/600/1500 ms) of a normal human brain. Original images with the desired and undesired ROI's are shown to the left, and the gray matter eigenimage is shown to the right. Signature and weighting vectors are graphed at the bottom. For the mathematical basis of the eigenimage filtering, see Section III-E.

vectors corresponding to voxels containing either pure white matter or a combination of 50% white matter and 50% gray matter are considered. For each case, the Euclidean distance between the mean vectors from the two categories is calculated and used in the calculation of the corresponding P_F. We do not need to consider the partial volumes between white matter and CSF, or gray matter and CSF since for these cases, the P_F's are smaller than those in Table I. This is because the Euclidean distances between the corresponding signature vectors are larger.

VI. Application and Evaluation

We have applied the new restoration filter as a preprocessing step for eigenimage filtering [2]–[4], [26]. Fig. 3 is a schematic representation of eigenimage filtering. Four multiple spin-echo and one inversion recovery image are shown to the left. The regions of interest for white matter, gray matter, and CSF are also shown. The eigenimage generated by taking gray matter as the desired feature, and white matter and CSF as the interfering features, are shown to the right. Here, the quality (CNR) of the eigenimage is good since in addition to four multiple spin-echo images, an inversion recovery image is also used. The white matter eigenimage using images 1–5 in Fig. 3 is shown in Fig. 5(i). However, using only four multiple spin-echo images, the eigenimage for white matter (shown in Fig. 5(e)) is so noisy that no structure can be seen in it. In this case, use of the new restoration filter as a preprocessing step before the eigenimage filtering has generated a highly improved eigenimage, which is shown in Fig. 5(h). This eigenimage shows the white matter structure clearly. It has segmented the desired feature (white matter) from the interfering features (gray matter and CSF) with a high CNR.

A. Examples

For experimental evaluation of the new filter as a preprocessing filter and comparing it with several conventional pre or postprocessing filters, we have used MRI scene sequences of a computer simulation and a human brain. Each image sequence consists of four multiple spin-echo images with TE/TR of 25, 50, 75, and 100/2500 ms. The simulated scene consists of several elliptic regions with different sizes (ellipses on the top simulate white matter, and those on the left simulate gray matter) and a square region (the central square simulates white matter, and L-shaped regions on the sides simulate CSF and gray matter) to illustrate effects of each filter on objects with different shapes and sizes. Partial volume regions between

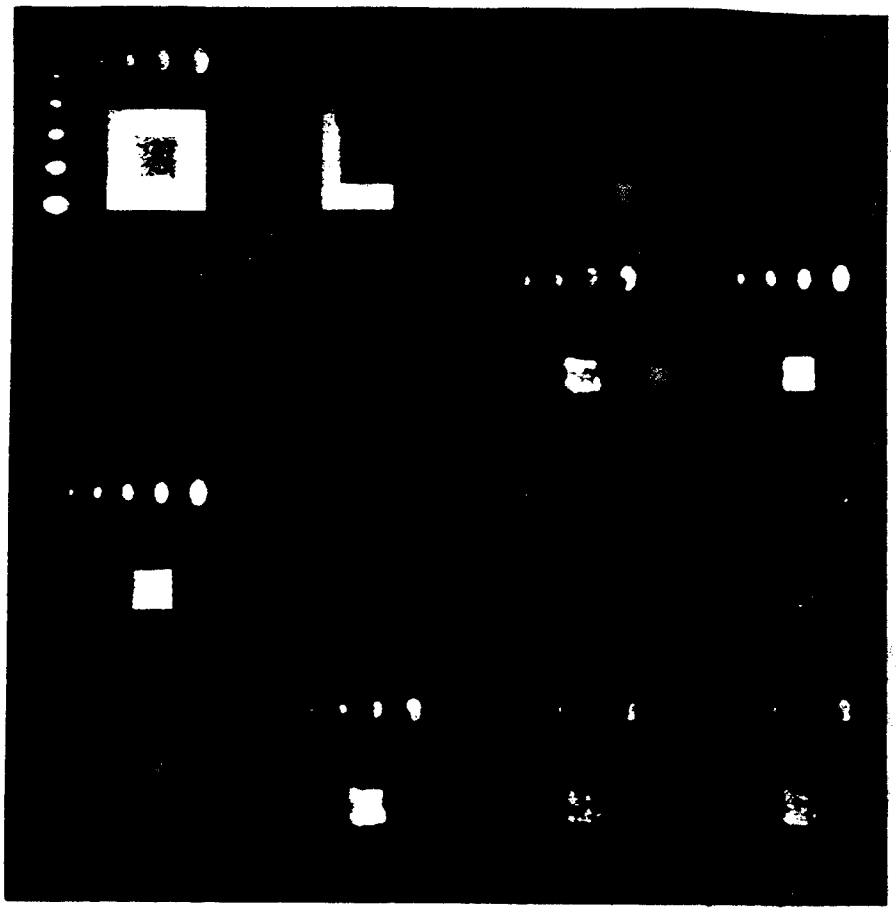


Fig. 4. (a)–(d) Four simulated spin-echo T2-weighted (TE/TR = 25–100/2500 ms) MR images. The scene consists of 12 ellipses, with different sizes and orientations, and a square with three overlapping regions; ellipses at the top and central region of the square represent white matter, ellipses on the left and the top and right strips of the square represent gray matter, and the left and bottom strips of the square represent CSF. Voxels corresponding to the pixels in the overlapping regions are assumed to contain convex combinations of the overlapping tissues. Images are windowed together to visualize the exponential decaying behavior of the MRI signal in each ROI; (e)–(h) eigenimages generated for the central region using the images shown in the first row and their preprocessed versions by MLE, AVG, and MLE + AVG filters using a threshold value of 4σ ($\sigma = 8.75$) and a neighborhood size of 9×9, respectively; (i) eigenimage generated using noiseless original images; (j)–(k) eigenimages generated using median and median-AVG preprocessed images; (l)–(p) post-processed eigenimages by low-pass (smooth 3-1 and G-9, as explained in Section VI-C), AVG, median, and median-AVG filters, respectively.

white matter and gray matter, and between white matter and CSF, are located on top and on the right-hand sides of the central square and on the left and bottom sides of it, respectively. Original images for the simulation and brain are shown in Figs. 4(a)–(d) and 5(a)–(d), respectively. Eigenimages without any pre or postprocessing are shown in Figs. 4(e) and 5(e), respectively. Eigenimages generated after the application of the MLE + AVG filter as a preprocessing step are shown in Figs. 4(h) and 5(h). Note the great improvement in the quality (CNR) of these eigenimages.

B. Effects of Neighborhood Size and Threshold Value

To illustrate the effects of neighborhood size and threshold value, we have generated several eigenimages for each example. These eigenimages are shown in Figs. 6 and 7. Neighborhood sizes for images in the first, second, third, and fourth rows are 3×3, 5×5, 7×7, and 9×9, respectively. Threshold values for the first, second, third, and fourth columns are 1σ ($\sigma = 8.75$), 2σ, 4σ, and 8σ, respectively. Table II lists CNR_1 (CNR between the desired feature and first interfering feature) and CNR_2 (CNR between the desired feature and second interfering feature) for each example, using several neighborhood sizes and threshold values.

It is seen that by increasing the threshold value from 1σ to 4σ, the CNR's of the eigenimage rapidly increase, but further increments of the threshold value do not considerably increase these CNR's. This is because increasing the threshold increases the probability of detection, which reaches 0.90 at 4σ (95% of possible improvement is achieved at this threshold level since CNR is proportional to the square root of the number of pixels averaged). Further increments of the threshold value may slightly improve the CNR's at the cost of increasing the probability of false alarm (inclusion of partial volume pixels that smooth edges and blur the whole image, as visualized by eigenimages in the third and fourth columns of Figs. 6 and 7). It is also observed that increasing the neighborhood size improves the CNR. This is because increasing the neighborhood size increases the number of pixels in each averaging step. This improvement is achieved at the price of increasing the computation time. Using a Sun SPARCstation 2, the computation time for 3×3, 5×5, 7×7, and 9×9 neighborhoods are approximately 0.4, 1.2, 2.5, and

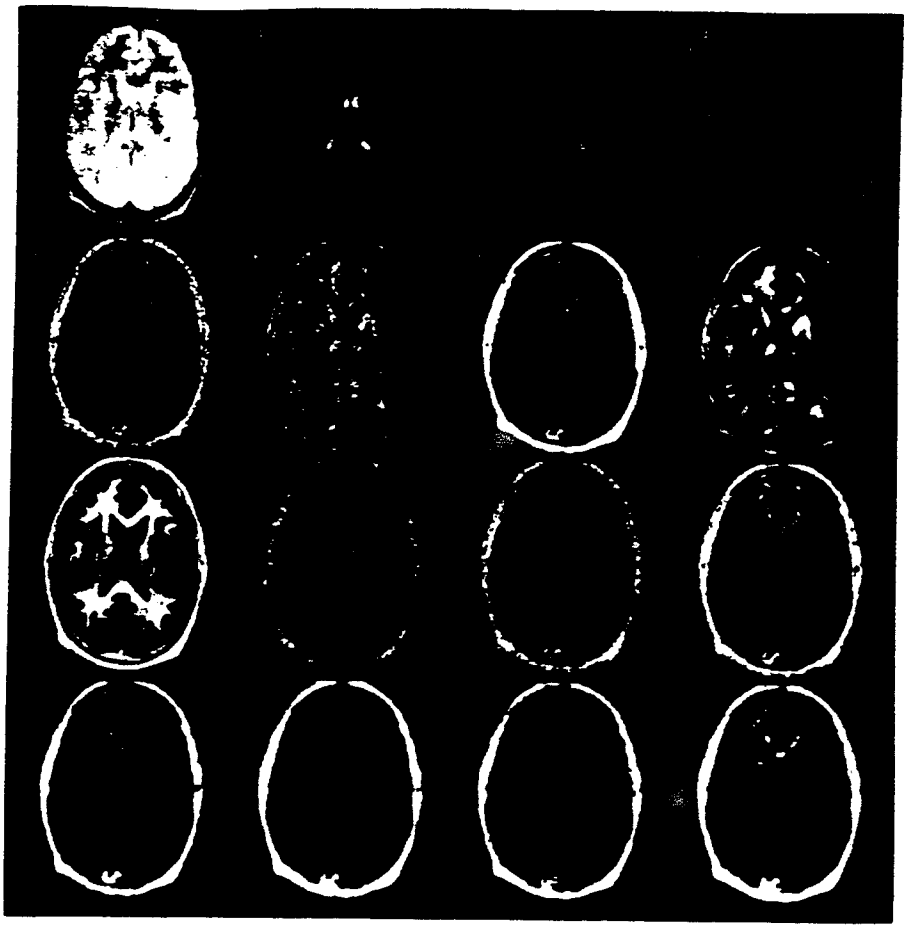


Fig. 5. (a)–(d) Four spin-echo T2-weighted (TE/TR = 25–100/2500 ms) MR images of a human brain. Images are windowed together to visualize the exponential decaying behavior of the MRI signal in each ROI; (e)–(h) eigenimages generated for the white matter using the images shown in the first row and their preprocessed versions by MLE, AVG, and MLE+AVG filters using a threshold value of 4σ ($\sigma = 8.75$) and a neighborhood size of 9×9, respectively; (i) eigenimage generated using the multiple spin-echo images in (a)–(d) plus an inversion recovery image; (j)–(k) eigenimages generated using median and median-AVG preprocessed images; (l)–(p) postprocessed eigenimages by low-pass (3-1 and G-9, as explained in Section VI-C), AVG, median, and median-AVG filters, respectively.

4.0 min, respectively. Since the filter is highly parallelizable, use of a parallel processing computer will significantly reduce the computation time. A Kalman filtering implementation of the MLE will also reduce the computation time.

Details of these implementations are beyond the scope of this paper. However, since the Kalman filtering has been conventionally applied to image restoration in a totally different context, a brief explanation of our intended use seems beneficial. For the proposed restoration filter, the MLE finds the minimum mean square fit to the 1-D signals (pixel vectors). An advantage of the Kalman filtering implementation of the minimum mean square solution is that it is faster than the original implementation [17]. A simple explanation for its speed is that at each point, it uses only the previous data points instead of using all of the data points. This increases the speed in a tradeoff with performance.

Images in Figs. 6–7, CNR's in Table II, and P_D's and P_F's in Table I illustrate that for a multiple spin-echo sequence with four echoes, a threshold value of 4σ and a neighborhood size of 9×9 are good choices. For this threshold level, about 90% of the pure pixels form the same tissue and only about 10% (on average) of those with less than 50% partial volume of the same tissue are used in each averaging step of the filter. These parameters were used to generate the results shown in Figs. 4(g)–(h), 5(g)–(h), and 8(c)–(d).

C. Comparison with Conventional Pre or Postprocessing Filters

Several other pre or postprocessing filters (some nonlinear and some linear) are also applied to the above examples. Nonlinear filters applied are vector median and vector median-average filtering [29] of the original images and median and median-average filtering of the eigenimages with a 9×9 neighborhood. Linear filters applied are low-pass filtering (spatial smoothing) with an impulse response of zero everywhere except for a square of 3×3 centered at the origin where it is one (smooth 3-1) or except for a square of 9×9, where it is proportional to a 2-D zero-mean Gaussian pdf with a variance of 4 (smooth G-9) of the eigenimages. The variance of the smooth G-9 filter was selected to implement an example of low-pass filtering with no significant frequency domain side lobes. The Gaussian kernel decreases from a maximum value of one at the center of the neighborhood to 0.13 at the edge centers and 0.02 at the corners of the 9×9

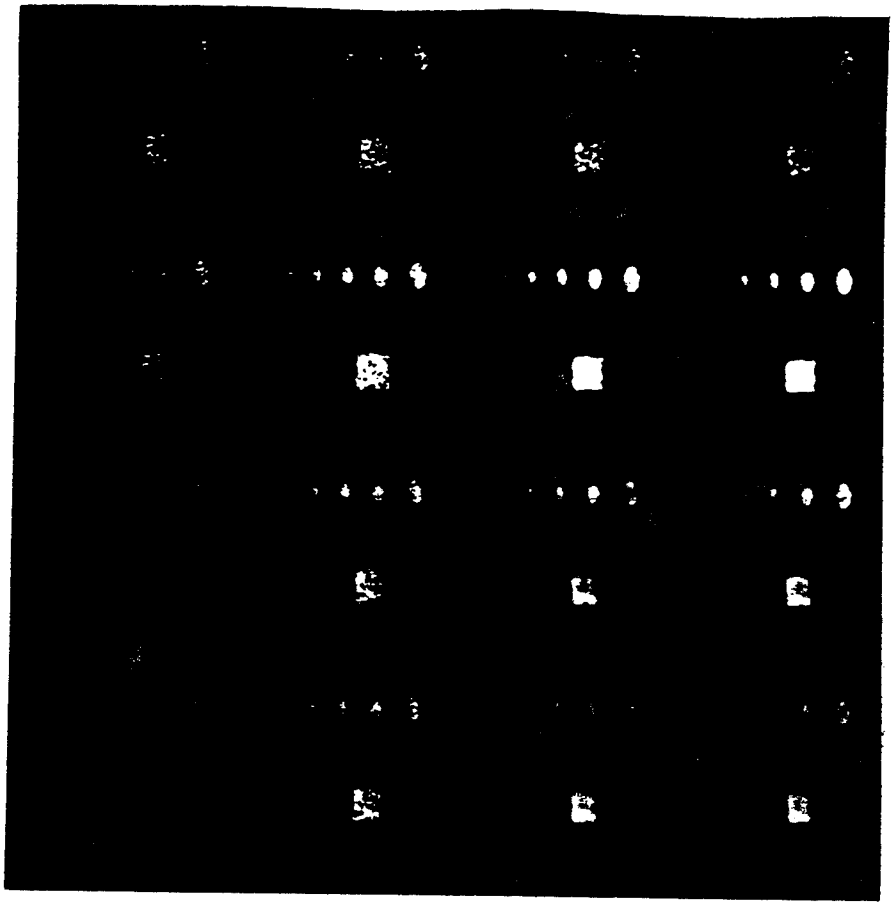


Fig. 6. Eigenimages generated for the central region of the simulation (which simulates white matter) after preprocessing (MLE + AVG) of the original images with several neighborhood sizes and threshold values. Neighborhood sizes for images in the first, second, third, and fourth rows are 3 × 3, 5 × 5, 7 × 7, and 9 × 9, respectively. Threshold values for the first, second, third, and fourth columns are 1σ ($\sigma = 8.75$), 2σ, 4σ, and 8σ, respectively.

neighborhood. Since eigenimage filter is linear, preprocessing of the original images by a linear filter is equivalent to postprocessing of the eigenimage by the same filter, except that preprocessing is computationally more intense. We therefore consider postprocessing of the eigenimages by linear filters. The resulting eigenimages, using the above filters, are shown in Figs. 4(i)–(p) and 5(j)–(p).

The MLE + AVG filter with a neighborhood size of 1 × 1 (MLE since no averaging is performed using this neighborhood) as well as without performing MLE (AVG) are also applied to each example. These runs show the effects of ignoring *intraframe (spatial)* or part of *interframe (parametric or temporal)* information, respectively. The corresponding eigenimages are shown in Figs. 4(f)–(g) and 5(f)–(g). The CNR's of the original images, restored images (MLE + AVG), and eigenimages discussed above are listed in Table III.

From eigenimages in Figs. 4 and 5 and their CNR's in Table III, it is seen that the MLE + AVG filter, which uses both interframe and intraframe information, outperforms all of the other methods. For the above examples, it generated an improvement in the CNR by a factor between 15 and 60 while preserving and enhancing edges. Moreover, as illustrated by brain eigenimages, other filters drop out inner regions of white matter with partial volume averaging effects. (This deficiency is not quantified by CNR since it is calculated using pure regions.) Impacts of each pre or postprocessing filter on the edge and outer regions with partial volume averaging effects are illustrated in the simulation example, for which the truth is known (the noiseless eigenimage is shown in Fig. 4(i)). It is clearly seen that the new restoration filter has preserved all of the edges in each ellipse, regardless of its size.

D. Effects of New Filter on Edge and Partial Volume Information

For further illustration of preserving edge and partial volume information by the AVG version of the new restoration filter while suppressing the additive noise (improving the SNR and CNR), we have performed the following experiments.

The first experiment uses a 1-D signal extracted from the simulation study by considering all pixels on the 170th row of the third simulated image shown in Fig. 4(c). The extracted signals from the noiseless simulation, from the simulation with an additive zero-mean white Gaussian noise, from the restored image using MLE + AVG filter, and from the restored image using AVG filter are shown in Fig. 8(a)–(d), respectively. They graphically illustrate that the additive noise is suppressed, edges are preserved, and partial volume information (slopes) is restored.

The second experiment uses an MRI scene sequence (four multiple spin-echo T2-weighted images with TE/TR = 25, 50,

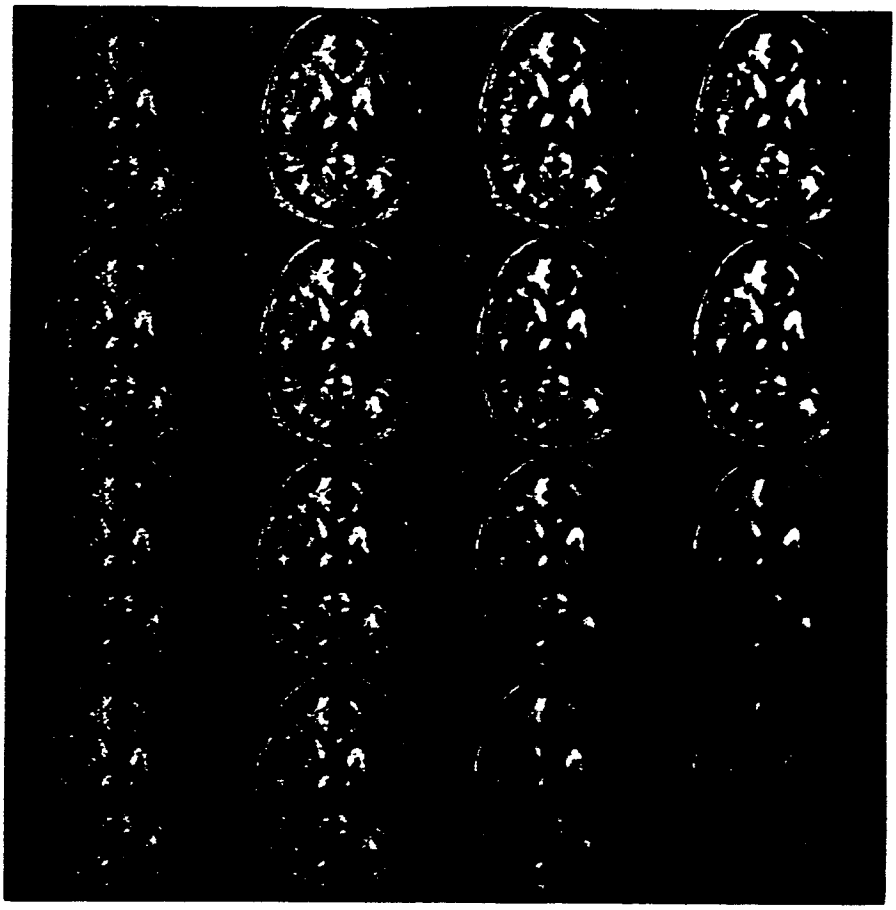


Fig. 7. White matter eigenimages generated after preprocessing (MLE + AVG) of the original images with several neighborhood sizes and threshold values. Neighborhood sizes for images in the first, second, third, and fourth rows are 3 × 3, 5 × 5, 7 × 7, and 9 × 9, respectively. Threshold values for the first, second, third, and fourth columns are 1σ ($\sigma = 8.75$), 2σ, 4σ, and 8σ, respectively.

75, 100/2500 ms and a spin-echo T1-weighted image with TE/TR = 20/500 ms) of a hard-boiled egg in gelatin. In this experiment, we have numerically estimated volumes of the egg white and egg yolk using the eigenimage filtered MR images[6] with and without the application of the AVG restoration filter. Due to the zero-mean property of the additive noise, the estimated volume, using many pixels in each slice and several slices through the object, should be close to the true volume unless the partial volume information is lost by the restoration filter while attenuating the additive noise. Thus, a comparison between the estimated volumes using the original images with those using the restored images may serve as an evaluation method for preserving the partial volume information. Table IV compares these estimated volumes with their actual volumes, which are measured from egg white and egg yolk water displacements. Closeness of the results, with and without the application of the restoration filter, indicates the preservation of the partial volume information.

Table IV also compares the estimated volume of the central region in the simulation using noiseless, noisy, and restored eigenimages. Closeness of the results again indicates the preservation of the partial volume information by the AVG version of the new filter.

[6] See [30] for a detailed explanation of the volume determination using the eigenimage filter.

VII. Summary and Conclusion

We presented the development and an application of a multidimensional restoration filter for MRI scene sequences. The proposed filter uses both interframe (parametric or temporal) and intraframe (spatial) information to filter the additive noise. Its performance depends on two sets of parameters: 1) data parameters such as noise power, CNR's between tissue types, and the number of images in the sequence and 2) filter parameters such as the neighborhood shape and size and the threshold value. Typical values of the data parameters, which were extracted from clinical MRI brain studies, were used to optimize the filter parameters. In particular, the threshold value was optimized based on the probabilities of detection and false alarm. Without *a priori* knowledge of the shapes of the objects in the scene, the neighborhood shape was chosen to be square. The computation time was the main factor in selecting the neighborhood size; it was decided that a 9 × 9 neighborhood was appropriate.

Since the new filter is specifically designed for MRI, it outperforms conventional methods, including median, median-average, spatial smoothing, and low-pass filtering. Application of the MLE + AVG version of the new filter as a preprocessing step before the eigenimage filtering improved the CNR's of the resulting eigenimages by a factor between 15 and 60 in a reasonable computation time (3 to 4 min on a Sun

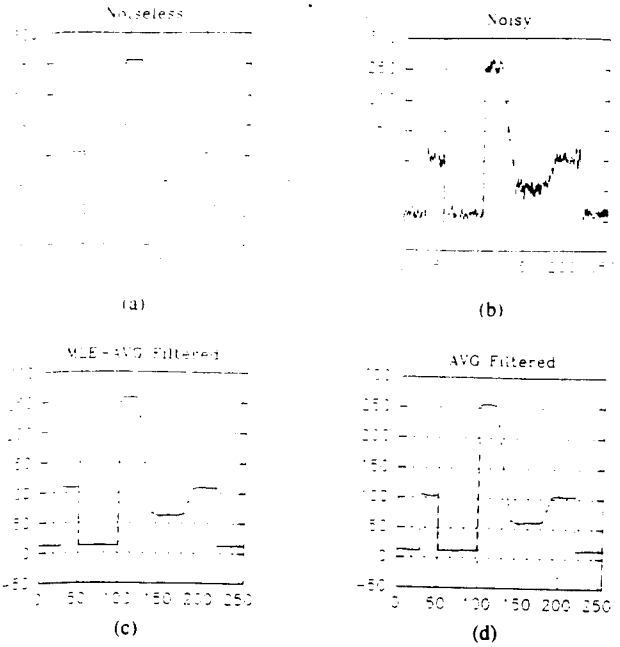


Fig. 8. One-dimensional signal extracted from the simulation study by considering all pixels on the 170th row of the third original image (Fig. 4(c)): (a) Without noise; (b) with zero-mean Gaussian noise; (c) restored using MLE + AVG filter; (d) restored using AVG filter. For both filters, a threshold value of 4σ ($\sigma = 8.75$) and a neighborhood size of 9×9 was used.

TABLE II
CNR's for the MLE + AVG Preprocessed Eigenimages of the Simulation and Brain, Using Several Neighborhood Sizes (NS) and Threshold Values (η)

		CNR_1		CNR_2	
NS	η	Simulation	Brain	Simulation	Brain
3×3	1σ	3.07	3.81	3.28	3.84
3×3	2σ	3.78	5.37	4.07	5.28
3×3	4σ	12.72	8.49	14.44	7.82
3×3	6σ	14.09	8.23	15.93	7.56
3×3	8σ	14.08	7.75	15.91	7.08
5×5	1σ	3.11	3.87	3.33	3.91
5×5	2σ	5.11	6.40	5.58	5.93
5×5	4σ	23.93	11.36	27.80	12.04
5×5	6σ	25.91	10.24	30.06	12.02
5×5	8σ	25.90	9.05	30.02	10.15
7×7	1σ	3.19	3.96	3.42	4.00
7×7	2σ	6.24	7.07	6.86	6.74
7×7	4σ	37.61	15.22	43.22	16.75
7×7	6σ	40.55	13.80	46.10	18.72
7×7	8σ	40.61	12.05	46.09	16.35
9×9	1σ	3.28	4.11	3.53	4.17
9×9	2σ	7.01	7.45	7.77	7.53
9×9	4σ	52.64	17.64	58.68	22.60
9×9	6σ	55.22	16.84	58.83	27.63
9×9	8σ	54.68	15.23	57.68	26.94

SPARCstation 2) while preserving and enhancing edges. The computation time may significantly be reduced by using a parallel processor and/or by a Kalman filtering implementation of the MLE. Details of these implementations were beyond the scope of this paper. As explained in Section VI-B, our intended use of the Kalman filter will increase the speed in a tradeoff

TABLE III
CNR's for the Original and Restored Images, and Preprocessed or Postprocessed Eigenimages of the Simulation and Brain

		CNR_1		CNR_2	
		Simulation	Brain	Simulation	Brain
Original	1st MSEI	-9.15	-7.85	-5.15	-4.52
images	2nd MSEI	-6.88	-7.70	-16.64	-20.39
and	3rd MSEI	-4.54	-5.09	-20.30	-21.64
eigen-	4th MSEI	-3.08	-3.81	-18.52	-24.19
images	Eigenimage	0.94	1.28	0.96	1.56
Restored	1st MSEI	-121.88	-14.79	-72.53	-14.09
images	2nd MSEI	-195.51	-16.71	-435.62	-78.12
using	3rd MSEI	-140.14	-16.94	-407.42	-127.71
MLE+AVG	4th MSEI	-114.24	-16.66	-333.06	-131.79
Eigen-	MLE	3.05	3.77	3.26	3.81
images	AVG	17.06	19.37	14.40	21.17
with	MLE+AVG	52.64	17.64	58.68	22.60
pre-	Median	0.93	1.76	0.88	2.04
processing	Median-AVG	10.95	10.35	10.18	12.08
Eigen-	Smooth 3-1	2.92	4.94	3.09	5.14
images	Smooth G-9	7.84	10.99	7.92	14.08
with	AVG	19.01	8.11	15.16	18.28
post-	Median	1.11	2.78	1.01	7.96
processing	Median-AVG	10.30	8.61	9.80	24.02

TABLE IV
A Comparison of Estimated and Actual Volumes for Egg White and Egg Yolk in an Egg Phantom and the Central Region (White Matter) in the Simulation. All Volumes Are Quoted in Cubic Centimeters

egg phantom	using water displacement	using acquired images	using AVG pre-processed images
egg-white's volume	34.8	34.3	34.2
egg-yolk's volume	21.8	21.7	21.3
simulation	**using noise-less images**	**using noisy images**	**using AVG pre-processed images**
central region's volume	58.13	57.19	58.68

with performance and is in a totally different context than that conventionally applied to image restoration.

Experimental results showed that the AVG version of the new restoration filter preserved partial volume information. These results indicate that the new multidimensional filter is successful in MR image restoration. It can be used as a preprocessing step to greatly improve the performance of image combination techniques, pattern recognition and classification algorithms, and tissue characterization methods. In a clinical study of human cerebral infarcts, it improved detectability of anatomical structures and tissue characteristics [31].

Between the two versions of the filter (MLE + AVG and AVG) with and without using the MRI signal models, the MLE + AVG version is most appropriate when processing images acquired using only one pulse sequence, e.g., a multiple spin-echo sequence, and thin slices. This is because 1) to our knowledge, there is no *simple* signal model that accurately fits a combination of different pulse sequences, e.g., a multiple spin-echo and an inversion recovery, and 2) for thick slices, a *simplified* signal model, e.g., the one given in (2), generates partial volume artifacts for overlapping tissues with significantly different intrinsic tissue parameters, e.g., white matter and CSF. On the other hand, the AVG version may be used for any MRI scene sequence acquired using an arbitrary

combination of pulse sequences. An important point, however, is that if the noise level changes from image to image, appropriate scales should be considered in the restoration filter, i.e., a scaled version of the Euclidean distance to be used, or images should be scaled before the application of the restoration filter.

Although this paper focused on the development and application of the restoration filter for MRI, its idea and theoretical bases are not limited to MRI. In particular, variations of the proposed filter can be applied to images other than MRI. We have successfully applied them to the computed tomography (CT), nuclear medicine (NM), and synthetic aperture radar (SAR) images. A manuscript explaining these applications and issues associated with them is in preparation.

ACKNOWLEDGMENT

The authors would like to thank L. Bower and S. Ramsey for their programming skills. They are also grateful to the associate editor and anonymous reviewers whose comments improved presentation of the paper.

REFERENCES

[1] J. B. de Castro *et al.*, "MR subtraction angiography with matched filter," *J. Comput. Assist. Tomogr.*, vol. 12, no. 2, pp. 355–362, 1988.

[2] J. P. Windham, M. A. Abd-Allah, D. A. Reimann, J. W. Froelich, and A. M. Haggar, "Eigenimage filtering in MR imaging," *J. Comput. Assist. Tomogr.*, vol. 12, no. 1, pp. 1–9, 1988.

[3] H. Soltanian-Zadeh, J. P. Windham, and J. M. Jenkins, "Error propagation in eigenimage filtering," *IEEE Trans. Med. Imaging*, vol. 9, no. 4, pp. 405–420, 1990.

[4] H. Soltanian-Zadeh, J. P. Windham, D. J. Peck, and A. E. Yagle, "A comparative analysis of several transformations for enhancement and segmentation of magnetic resonance image scene sequences," *IEEE Trans. Med. Imaging*, vol. 11, no. 3, pp. 302–318, 1992.

[5] E. R. McVeigh, R. M. Henkelman, and M. J. Bronskill, "Noise and filtration in magnetic resonance imaging," *Med. Phys.*, vol. 12, no. 5, pp. 586–591, 1985.

[6] F. W. Wehrli, "Signal-to-noise and contrast in MR imaging," in *NMR in Medicine, the Instrumentation and Clinical Applications* (S. R. Thomas and R. L. Dixon, Eds.). Amer. Assoc. Phys. Medicine, Medical Physics, 1986, pp. 216–228, Monograph No. 14,

[7] R. E. Hendrick, "Image contrast and noise," in *Magnetic Resonance Imaging*. St. Louis: Stark and Bradly, Mosby-Year Book, 1988, pp. 66–83, 1st ed., ch. 5.

[8] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Patt. Anal. Macine Intell.*, vol. 12, no. 7, pp. 629–639, 1990.

[9] P. Saint-Marc, J. S. Chen, and G. Medioni, "Adaptive smoothing: A general root for early vision," in *Proc. IEEE Comput. Soc. Conf. Comput. Vision Patt. Recogn.*, 1989, pp. 618–624.

[10] A. C. Bovik, T. S. Huang, and D. C. Munson, "A generalization of median filtering using linear combination of order statistics," *IEEE Trans. Acoust, Speech, Signal Processing*, vol. ASSP-31, no. 12, pp. 1342–1350, 1983.

[11] J. B. Bendar and T. L. Watt, "Alpha trimmed means and their relationship to median filters," *IEEE Trans. Acoust, Speech, Signal Processing*, vol. ASSP-32, no. 2, pp. 145–153, 1984.

[12] Y. H. Lee and S. A.Kassam, "Generalized median filtering and related nonlinear filtering techniques," *IEEE Trans. Acoust, Speech, Signal Processing*, vol. ASSP-33, no. 3, pp. 672–683, 1985.

[13] R. Ding and A. N. Venetsanopoulos, "Generalized homomorphic and adaptive order statistic filters for the removal of impulsive and signal-dependent noise," *IEEE Trans. Circ. Syst.*, vol. CAS-34, no. 8, pp. 948–955, 1987.

[14] W. B. McCain and C. D. McGillem, "Performance improvement of DPLL's in non-gaussian noise using robust estimators," *IEEE Trans. Acoust, Speech, Signal Processing*, vol. ASSP-35, no. 11, pp. 1207–1216, 1987.

[15] A. Restrepo and A. C. Bovik, "Adaptive trimmed mean filters for image restoration," *IEEE Trans. Acoust, Speech, Signal Processing*, vol. 36, no. 8, pp. 1326–1337, 1988.

[16] M. Pietikainen and D. Harwood, "Segmentation of color images using edge-preserving filters," in *Advances in Image Processing and Pattern Recognition* (V. Cappellini and R. Marconi, Eds.). Amsterdam: Elsevier, North Holland, 1986, pp. 94–99.

[17] H. L. Van Trees, *Detection, Estimation, and Modulation Theory*. New York: Wiley, 1968.

[18] H. Soltanian-Zadeh, J. P. Windham, and D. O. Hearshen, "Preprocessing of MR image sequences using a new edge-preserving multi-dimensional filter." Presented at the 10th Ann. Mtg. Soc. Mag. Res. Med. (SMRM), San Francisco, CA, Aug. 10–16, 1991 (Abstract published in the SMRM Book of Abstracts, vol. 2, p. 748, 1991).

[19] J. P. Windham, A. M. Haggar, D. O. Hearshen, J. R. Roebuck, and D. A. Reimann, "A novel method for volume determination using MR image sequence," Presented at the 7th Ann. Mtg. Soc. Mag. Res. Med. (SMRM), San Francisco, CA, Aug. 20–26, 1988 (Abstract published in the SMRM Book of Abstracts, vol. 2, pp. 1081, 1988).

[20] H. Stark and J. W. Woods, *Probability, Random Processes, and Estimation Theory for Engineers*. Englewood Cliffs: Prentice-Hall, 1986.

[21] J. N. Lee and S. J. Riederer, "The contrast-to-noise in relaxation time, synthetic, and weighted-sum MR images," *Mag. Res. Med.*, vol. 5, pp. 13–22, 1987.

[22] D. G. Brown *et al.*, "CNR enhancement in the presence of multiple interfering processes using linear filters," *Mag. Res. Med.*, vol. 14, no. 1, pp. 79–96, 1990.

[23] R. B. Buxton, "Target-point combination of MR images," *Magn. Res. Med.*, vol. 18, no. 1, pp. 102–115, 1991.

[24] S. L. Meyer, *Data Analysis for Scientists and Engineers*. New York: Wiley, 1975, pp. 39–48.

[25] M. A. Bernstein, D. M. Thomasson, and W. H. Perman, "Improved detectability in low SNR magnetic resonance images by means of phase-corrected real reconstruction," *Med. Phys.*, vol. 16, no. 5, pp. 813–817, 1989.

[26] H. Soltanian-Zadeh, J. P. Windham, and A. E. Yagle: "Optimal transformation for correcting partial volume averaging effects in magnetic resonance imaging." *IEEE Trans. Nuc. Sci.*, vol. 40, no. 4, pp. 1204–1212, 1993.

[27] S. M. Kay, *Modern Spectral Estimation, Theory and Application*. Englewood Cliffs: NJ: Prentice-Hall, 1988.

[28] J. L. Devore, *Probability and Statistics for Engineering and the Sciences*. Monterey, CA: Brooks/Cole, 2nd ed., 1987.

[29] P. Haavisto, P. Heinonen, and Y. Neuvo, "Vector FIR-median hybrid filters for multispectral signals," *Electron. Lett.*, vol. 24, no. 1, pp. 7–8, 1988.

[30] D. J. Peck, J. P. Windham, H. Soltanian-Zadeh, and J. Roebuck, "A fast and accurate algorithm for volume determination in MRI," *Med. Phys.*, vol. 19, no. 3, pp. 599–605, 1992.

[31] J. P. Windham, H. Soltanian-Zadeh, and D. J. Peck, "Delineation of internal structure in cerebral tumors using MRI," Presented at the 34th Ann. Mtg. Amer. Assoc. Phys. Medicine (AAPM), Calgary, Canada, Aug. 1992 (Abstract published in Med. Phys., vol. 19, no. 3, p. 844, 1992).

Hamid Soltanian-Zadeh (S'89–M'92) was born in Yazd, Iran, in 1960. He received the B.S. and M.S. degrees in electrical engineering: electronics from Tehran University, Tehran, Iran, in 1986 and the M.S.E. and Ph.D. degrees in electrical engineering: Systems from the University of Michigan, Ann Arbor, in 1991 and 1992, respectively.

From 1985 to 1986, he was with Iran Telecommunication Research Center in Tehran. In 1987, he was a lecturer of electrical engineering at Tehran University. Since September 1988, he has been with the Department of Diagnostic Radiology and Medical Imaging of Henry Ford Hospital, Detroit, MI, where he is currently a Senior Staff Scientist. He has also been with the Department of Electrical Engineering and Computer Science of the University of Michigan, where he currently holds a Visiting Scholar affiliation. His research interests include medical imaging, image reconstruction and processing, pattern recognition, and neural networks.

Joe P. Windham (M'92) received the Ph.D. degree in nuclear science and engineering from the University of Cincinnati, OH, in 1972.

He was at the Medical College of Ohio, Toledo, from 1972 to 1984, where he was an Associate Professor and Head of the Division of Medical Physics of the Department of Radiology. He has been with the Department of Radiology of Henry Ford Hospital, Detroit, MI, since June 1984, where he is currently Head of the Division of Radiological Physics. He is certified by the American Board of Radiology in Radiological Physics. His research interests include image processing and quantitative analysis of medical images particularly images obtained from magnetic resonance, computed tomography, and nuclear medicine.

Dr. Windham is active in the American College of Radiology, where he is a member of the Commission on Physics and Radiation Safety, Commission on Education, and Commission on Radiation Oncology. He is chairman of the Committee on Education under the Commission on Physics and Radiation Safety.

Andrew E. Yagle (M'85) was born in Ann Arbor, MI, in 1956. He received the B.S.E. and B.S.E.E. degrees from the University of Michigan, Ann Arbor, in 1977 and 1978, respectively, and the S.M., E.E., and Ph.D. degrees from the Massachusetts Institute of Technology (MIT), Cambridge, in 1981, 1982, and 1985, respectively.

While at MIT, from 1982 to 1985, he was on an Exxon Teaching Fellowship. Since September 1985, he has been with the Department of Electrical Engineering and Computer Science, the University of Michigan, Ann Arbor, where he is currently an Associate Professor. His research interests include fast algorithms for digital signal processing, multiresolution and iterative algorithms in medical imaging, multidimensional inverse scattering, phase retrieval, and linear least-squares estimation.

Dr. Yagle received the NSF Presidential Young Investigator Award in 1988 and the ONR Young Investigator Award in 1990. He received H.H. Rackham School of Graduate Studies Research Partnership Awards with T.-S. Pan in 1990 and with K.R. Raghavan in 1993. He has received several teaching awards, including the College of Engineering Teaching Excellence Award in 1992, the Eta Kappa Nu Professor of the Year Award in 1990, and the Class of 1938e Distinguished Service Award in 1989. He is currently an Associate Editor of the IEEE SIGNAL PROCESSING LETTERS, IEEE TRANSACTIONS ON IMAGE PROCESSING, and *Multidimensional Systems and Signal Processing*. He is also a member of the Digital Signal Processing Technical Committee, a past Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING, and co-technical chair of ICASSP-95, which will be held in Detroit, MI.

APPENDIX T

P. Raadhakrishnan, J. Dorband, and A.E. Yagle, "An Algorithm for Forward and Inverse Scattering in the Time Domain," submitted to J. Acoust. Soc. Am., September 1993.

As noted above, the invariant imbedding algorithm used to generate the forward problem data is computationally intensive. This is because invariant imbedding, although similar to layer stripping in approach, is quite different, in that it does not take advantage of time causality. Hence it solves many more problems than are actually needed. Layer stripping avoids this and is more efficient.

This paper is a first attempt at parallelizing invariant-imbedding-based algorithms for both the forward and the inverse problems, and thus reducing the computation required. Only the 1-D problem is considered here. The new algorithm is parallelizable, and hence requires less computation, and gives results identical to the invariant imbedding algorithm of Corones et al. An simple error analysis of the effects of computational noise on the algorithm is also supplied.

An Algorithm for Forward and Inverse Scattering in Time Domain

Raadhakrishnan Poovendran
HSTX/NASA/GSFC
Mail Code 934
Greenbelt MD 20771
rads@hartley.gsfc.nasa.gov

John E. Dorband
NASA/Goddard Space Flight Center
Mail Code 934
Greenbelt, MD 20771
dorband@nibbles.gsfc.nasa.gov

PACS numbers: 43.20.Fn 43.20.Bi

This paper presents an alternative numerical scheme for a class of forward and inverse scattering problems based on invariant imbedding method [1]. The algorithm presented in this paper is decoupled more than the previous implementations and hence, readily amenable to parallel processing. It is also shown that the effect of this decoupling on error can be expressed in terms of the estimated reflection coefficients, facilitating the decision to accept or reject any computed parameter.

1 Introduction

When a wave propagates through a medium with varying material properties, the velocity of the wave changes. The process of such wave propagation is called wave scattering. Many interesting scattering problems arise in the real world in physical processes such as speech processing, geophysics, acoustics, transmission line modeling, etc. The study of the scattering process is divided into two main categories: forward and inverse scattering. The forward scattering problem deals with the analysis of the wave propagation within a medium. The inverse problem deals with the reconstruction of the medium properties from the knowledge of a time sequence of reflections from the medium.

There are two major approaches for solving the scattering problems. One approach is based on the integral method such as Merchenko formulation, and the other is based on the differential layer stripping methods. Differential methods also lead to many interesting physical interpretations. In [8], a host of differential methods for solving inverse scattering were discussed and a unified framework was established for many algorithms. As noted in [8], for computational reasons, the most popular algorithms used to solve scattering problems are dynamic deconvolution and differential methods in time or frequency domains. The differential methods implicitly assume that the medium is inhomogeneous but sufficiently smooth. Smoothness assumption enables one to view the medium as a physical system consisting of multiple layers, across which the medium properties

vary in a smooth but inhomogeneous manner. By subdividing the smooth inhomogeneous medium into differential layers in which the medium properties remain constant, the necessary scattering equations are derived. Solutions to two adjacent layers are connected by the common boundary condition. In a differential setup, the inverse problem deals with the material reconstruction in a thin differential layer, whereas the forward problem deals with the construction of the reflection kernel differentially. By propagating the solution across the layers, the whole system is solved for the forward or the inverse problems.

The time domain forward and inverse scattering problems allow one to reconstruct the inhomogeneous medium locally [1] - [3]. In [1], an analytical method based on invariant imbedding ([3], [7]) was proposed to solve for the forward and inverse scattering problems in time domain. A non-linear differential integral equation was derived in terms of the reflection kernel of the inhomogeneous medium using invariant imbedding techniques. Because the reflection kernel is connected to the medium properties, computing the reflection kernel in a local region enables one to compute the material properties in that region. Hence, the same equation was used for the forward and the inverse scattering problems. In frequency domain problems, computation of the reflection kernel at any depth involves the knowledge of the reflection kernel at different depths, as can be seen from the derivation presented below.

This paper deals with a reformulation of a class of 1-D forward and inverse scattering problems, employing an invariant imbedding method. The paper is divided into five main sections. Section 2 presents the formulation of the problem, the necessary propagation equation, and the available computational algorithms [1] and their reduntancies. Section 3 presents an efficient reformulation of the existing computational algorithms to derive the modified algorithms. A quick derivation of the error bound for the modified forward problem is presented in section 4. Section 5 deals with the issues in numerical simulations of the modified forward and the inverse problems. For illustration purposes, the timings for the forward and the inverse problems are also presented.

2 Problem Description

In this section we describe the problem presented in [1] - [3]. (Refer to [1] - [2] for the details of the problem). An alternative brief derivation of the differential integral equations is presented below.

To derive the mathematical relationship between the reflection kernel and the material properties of the inhomogeneous medium, the following assumptions were made in [1]:

1. The material properties of the inhomogeneous region of the medium are sufficiently smooth and vary as a function of the depth z of the medium.
2. The medium is inhomogeneous in the interval $a_0 < z < b_0$ and homogeneous elsewhere.
3. The material properties for $z < a_0$ and $z > b_0$ match those at $z = a_0$ and $z = b_0$, respectively. Hence, the incident and the reflected wavefields can be easily identified.

Fig. 1 shows the velocity profile $c(z)$ of the wavefield. The velocity $c(z)$ in the inhomogeneous region of the medium ($a_0 < z < b_0$) is continuous at the boundaries a_0 and b_0.

The basic idea behind invariant imbedding methods is to subdivide the interval $[a_0, b_0]$ into n smaller intervals with varying widths, and solve the scattering equation for a thin layer with known boundary conditions. Using this solution, the scattering problem for the next layer can be solved and the solution can then be propagated to the other end. In doing so, one implicitly assumes that outside the boundaries of each thin layer, the medium is homogeneous, i.e., two layers interact only by the common boundary. Hence, the resulting equations are functions of the varying boundary coordinates. We derive the necessary non-linear differential integral equations presented in [1] based on the two-component model of the wave propagation [8].

2.1 Propagation Equation

The propagation equation is derived using the two-component frequency domain model in terms of the travel-time coordinate defined by ([1], [5], [8]):

$$x = \phi(z) = \int_{a_0}^{z} \frac{ds}{c(s)} \tag{1}$$

and

$$\frac{d}{dx}\begin{bmatrix} \hat{D} \\ \hat{U} \end{bmatrix} = \begin{bmatrix} -i\omega & r(x) \\ r(x) & i\omega \end{bmatrix}\begin{bmatrix} \hat{D} \\ \hat{U} \end{bmatrix}, \tag{2}$$

where $\hat{D} = \hat{D}(x,\omega)$ and $\hat{U} = \hat{U}(x,\omega)$ are the wave components along the positive and the negative x directions, respectively. Letting $\hat{R}(x,\omega) = \hat{U}(x,\omega)/\hat{D}(x,\omega)$ and differentiating $\hat{R}(x,\omega)$ with respect to x leads to the following equation:

$$\frac{d}{dx}\hat{R}(x,\omega) = 2i\omega\hat{R}(x,\omega) + r(x) - r(x)\hat{R}^2(x,\omega). \tag{3}$$

As mentioned in the introduction, the frequency domain equations involve the knowledge of the reflection kernel at all depths and hence local reconstruction is not favored. In order to solve the local properties, the time domain equation is derived by taking an inverse *Fourier* transform of equation (3). The associated boundary condition can be derived using the final value theorem.

$$\frac{\partial}{\partial x}R = 2\frac{\partial}{\partial t}R - r(x)(R(x,t-x) \otimes R(x,t-x)) + r(x)\delta(t) \tag{4a}$$

where $\otimes$ denotes the linear convolution in time variable.

$$R(x,y,0^+) = -\frac{r(x)}{2}; \qquad R(y,y,t) = 0; \qquad t > 0. \tag{4b}$$

where $A(x) = 2r(x)$. Equation (4a) is essentially the *Ricatti* equation derived in [1]. Travel time coordinate transformation can also be interpreted as the velocity normalization. Hence, equation (4a) can be further reduced to the following form:

$$\frac{\partial}{\partial x}R = -r(x)(R(x,t-x) \otimes R(x,t-x)) + r(x)\delta(t) \tag{5}$$

In the forward problem, we are given $A(x)$ for $0 < x < \Phi(b_0)$ and using equation (4a) we can solve for the reflection kernel of the medium, $R(0,\Phi(b_0),t) = R^+(a_0,b_0,t)$. In the inverse problem, we are given a finite initial section of the reflection kernel $R^+(a_0,b_0,t)$ for $0 < t < 2\Phi(b_0)$. ($2\Phi(b_0)$ is the round trip travel time for a pulse to go to the depth b_0 and return to the surface.) From the knowledge of this data, $A(x)$ is reconstructed for $0 < x < \Phi(b_0)$ in a differential manner. Starting with the knowledge of $R(x,y,t)$ at depth x, $A(x)$ is computed. Next, $R(x+\delta x,y,t)$ is computed by advancing one layer into the medium using equation (4a) and the knowledge of $A(x)$ at depth x. Solution of $R(x+\delta x,y,t)$ is then used to compute $A(x+\delta x)$. This recursive procedure is essentially the dynamic deconvolution process.

2.2 Computational Algorithms for the Ricatti Equation

Assuming that the travel-time coordinates are normalized to unity, we set $y = 1$ and represent $R(x,y,t)$ as $R(x,t)$. This is equivalent to fixing the left boundary of the problem. Furthermore, we set the computational grid for the problem to be $0 < x < 1$ and $0 < t < 2(1-x)$. In this setup, for the forward problem, $A(x)$ is known and $R(0,t)$ has to be computed for $0 < t < 2$. For the inverse problem, $R(0,t)$ is specified for $0 < t < 2$ and $A(x)$ has to be obtained for $0 < x < 1$.

Using the Trapezoidal rule and integrating over the interval $(x_0, x_0+\epsilon)$, equation (5) can be rewritten as:

$$R(x_0+\epsilon, t-2(x_0+\epsilon)) - R(x_0,t-2x_0) =$$

$$-\frac{\epsilon}{4}[A(x_0+\epsilon)R(x_0+\epsilon,t-2(x_0+\epsilon))\otimes R(x_0+\epsilon,t-2(x_0+\epsilon)) + A(x_0)R(x_0,t-2x_0)\otimes R(x_0,t-2x_0)]. \tag{6}$$

The next two subsections present the numerical schemes for the forward and the inverse scattering problems, respectively.

2.2.1 Numerical Algorithm for the Forward Problem

For the forward problem, the discrete boundary condition $R_{i,0} = -A_i/4$ is given, and we solve equation (6) for the reflection kernel $R_{i,j}$ by moving from right to left and bottom to top along the computational grid given in Fig. 2.

The numerical value of the reflection kernel at the circled point in Fig. 2 is computed from the values of the reflection kernel at the grid points marked **x**. The corresponding algorithm is given by:

- For $i = n$ to 0 Begin
- $R_{i,0} = -A_i/4$
- For $j = 0$ to $n - i$ Begin
- $R_{i,j} = (1 + \epsilon^2 A_i^2/8)^{-1} \left[R_{i+1,j-1} + \frac{\epsilon^2}{2} \left(A_{i+1} \sum_{k=1}^{k=j-1} R_{i+1,k} R_{i+1,j-1-k} + A_i \sum_{k=1}^{j-1} R_{i,k} R_{i,j-k} \right) \right]$
- End
- End.

2.2.2 Numerical Algorithm for the Inverse Problem

For the inverse problem, the discrete boundary condition $A_i = -4R_{i,0}$ is given. Equation (6) is then solved for the reflection kernel $R_{i,j}$ and A_i simultaneously by moving from left to right and top to bottom along the computational grid shown in Fig. 3.

The numerical value of the reflection kernel at the circled point in Fig. 3 is computed from the values of the reflection kernel at the grid points marked **x**. The corresponding algorithm is given by:

- $A_0 = -4R_{0,0}$
- For $i = 0$ to $n - 1$ Begin
- $R_{i,0} = R_{i,1} - \frac{\epsilon^2}{2} A_i R_{i,0} R_{i,1}$
- $A_i = -4R_{i+1,0}$
- For $j = 0$ to $n - i$ Begin
- $R_{i+1,j} = (1 - \epsilon^2 A_{i+1}^2/8)^{-1} \left[R_{i,j+1} + \frac{\epsilon^2}{2} \left(A_{i+1} \sum_{k=1}^{k=j-1} R_{i+1,k} R_{i+1,j-k} + A_i \sum_{k=1}^{j-1} R_{i,k} R_{i,j+1-k} \right) \right]$
- End

- End.

From equation (6) and the computational grids for the forward and the inverse problems, we note that the reflection kernel at depths x_0 and $x_0 + \epsilon$ or $x_0 - \epsilon$ appear on both sides. Hence, the computation of the reflection kernel at a given depth x requires some information about the reflection kernel at the same depth. Because of this coupling, it is preferable to modify equation (6) so that the computation of the reflection kernel at any given depth does not require the knowledge of any other point of the reflection kernel at the same depth. Such a modification can improve performance in a parallel computer and reduce error propagation due to the coupling of terms. In the next section we show that such a modification is indeed possible and straight forward.

3 Reformulation of the Computational Algorithms for the Riccatti Equation

Since the medium is assumed to be sufficiently smooth, the reflection kernel is a smooth function and, hence, a continuous function of depth x. For example, the reflection kernel $R(x_0+\frac{\epsilon}{2}, t-2(x_0+\frac{\epsilon}{2}))$ is left and right continuous. This leads to the following constraint:

$$\lim_{\epsilon\to 0}\frac{1}{\epsilon}\{R(x_0+\frac{\epsilon}{2}, t-2(x_0+\frac{\epsilon}{2})) - R(x_0, t-2x_0)\} =$$

$$\lim_{\epsilon\to 0}\frac{1}{\epsilon}\{R(x_0+\epsilon, t-2(x_0+\epsilon)) - R(x_0+\frac{\epsilon}{2}, t-2(x_0+\frac{\epsilon}{2}))\}. \tag{7}$$

In the computational algorithm described in the previous subsection, ϵ denotes the discrete step size. Since the reflection kernel is a smooth function of depth, instead of integrating equation (4a) over the interval $(x_0, x_0+\epsilon)$, we can integrate it over the intervals $(x_0, x_0+\frac{\epsilon}{2})$ and $(x_0+\frac{\epsilon}{2}, x_0+\epsilon)$. Using the results from equation (7), the following forward equation can be derived:

$$R_{i-1,j+1} = R_{i,j} + \frac{\epsilon^2}{4}[A(i+1)\sum_{k=1}^{j-1} R_{i+1,k}R_{i+1,j-1-k} + A(i)\sum_{k=1}^{j} R_{i,k}R_{i,j-k}]. \tag{8}$$

Clearly, the reflection kernel has to be computed for the $(i-1)^{st}$ layer, and index j is bounded above by $(n-i-1)$ instead of the conventional bound $n-i$.

Similarly, using equation (7), the discrete version of the inverse problem can be derived as:

$$R_{i+1,j-1} = R_{i,j} - \frac{\epsilon^2}{4}[A(i)\sum_{k=1}^{j} R_{i,k}R_{i,j-k} + A(i-1)\sum_{k=1}^{j+1} R_{i-1,k}R_{i-1,j-k}]. \tag{9}$$

We note that, unlike the algorithm in [1], the summation limits in equations (8) and (9) do vary at any layer level. Also, the modified equations allow one to compute the reflection kernel at any depth independently of the reflection kernel at that depth at a previous time. This decoupling enables one to compute the different time records of the reflection kernel at any fixed depth in a parallel manner. Moreover, any error propagation due to the computation of the different time records of the reflection kernels will not affect the reflection kernel at the same depth. Hence, the modified algorithm has slightly better control over the error propagation. Moreover, as detailed in the next two subsections, initialization of the modified computational algorithm also differs from the algorithm in [1].

3.1 Modified Numerical Algorithm for the Forward Problem

For the forward problem, the discrete boundary conditions $R_{i,0} = -A_i/4$ and $R_{n-1,1} = 2A_n/(8 + \epsilon^2 {A_n}^2)$ are used to initialize the algorithm. Equation (6) is then solved for the reflection kernel $R_{i,j}$ by moving along right to left and bottom to top along the computational grid shown in Fig. 4.

The numerical value of the reflection kernel at the circled point in Fig. 4 is computed from the values of the reflection kernel at the grid points marked **x**. It is important to note that the computation of the reflection kernel at the circled point does not require the knowledge of any other reflection kernel at the same layer. Due to the modification done on the forward equation, the computational algorithm needs an extra initial point $R_{n-1,1}$ as follows:

- $R_{n,0} = -A_n/4$;
- $R_{n-1,1} = 2A_n/(8 + \epsilon^2 {A_n}^2)$
- For $i = n - 1$ to 0 Begin
- $R_{i,0} = -A_i/4$
- For $j = 0$ to $n - i$ Begin
- $R_{i-1,j+1} = R_{i-1,j} + \frac{\epsilon^2}{4}[A(i+1)\sum_{k=1}^{j-1} R_{i+1,k}R_{i+1,j-1-k} + A(i)\sum_{k=1}^{j} R_{i,k}R_{i,j-k}]$
- End
- End.

3.2 Modified Numerical Algorithm for the Inverse Problem

In the case of the inverse problem, the reflection kernel on the surface $z = 0$ is available. For $z < a_0$, the medium is homogeneous, and to initialize the algorithm we add another layer in the homogeneous medium. Due to homogeneity, the reflection kernel of the added layer will be identical

to the time records of the reflection kernel of the surface. Reflection kernel at the circled point in Fig. 5 is computed from top to bottom and from left to right along the computational grid as shown below:

- $A_0 = -4R_{0,0}$
- For $i = 0$ to $n-1$ Begin
- $R_{i,0} = R_{i,1} - \frac{\epsilon^2}{2} A_i R_{i,0} R_{i,1}$
- $A_i = -4R_{i+1,0}$
- For $j = 0$ to $n-i$ Begin
- $R_{i+1,j-1} = R_{i,j} - \frac{\epsilon^2}{4}[A(i)\sum_{k=1}^{j} R_{i,k}R_{i,j-k} + A(i-1)\sum_{k=1}^{j+1} R_{i-1,k}R_{i-1,j-k}]$
- End
- End.

4 Error Propagation for the Forward Problem

The modified equations are not only decoupled, but also have the same order of accuracy as the existing algorithm. The error propagation can be tracked to the same order of accuracy. If, at layer number i_0 and location number j_0, an error of size λ is made in the forward algorithm, it can be shown that the error generated in computing $R_{i,j}$ is given by:

$$\tilde{R}_{i,j} = \lambda \left[\delta(j - j_0 - i_0 + i) - \left(2\alpha \sum_{l=i_0}^{i+1} A(l) R_{i,j-j_0-i_0+i} + \alpha A(i+1) R_{i,j-j_0-i_0+i} \right) U(j - j_0 - i_0 + i) \right], \quad (10$$

after ignoring the third and the higher order terms of ϵ. If $A(x)$ and $R_{i,j}$ are bounded above by M and p respectively, the upper bound on the error at point $R_{i,j}$ due to a computational error of strength λ at R_{i_0,j_0} is given by:

$$\tilde{R}_{i,j} = \lambda \left[1 + \frac{\epsilon^2}{2} Mp(i - i_0) + 3\frac{\epsilon^2}{4} Mp \right], \quad (11)$$

where $\tilde{R}$ denotes the error contribution. A similar bound can be derived for the inverse algorithm.

5 Numerical Examples

In this section we describe the numerical simulations performed using the modification presented in this paper.

Simulations were aimed at testing the following issues:

- Check the validity of the reformulation for the sequential version given a fixed discretization;
- Implement a completely parallel algorithm for the forward and inverse problems;
- Compare the computational timings for variable layer numbers for the sequential and parallel algorithms.

Next section details the initializations made in the computation.

5.1 Initializations

Due to the modification made in the forward algorithm, initialization of the computation requires information of three initial points: $R_{n,0}$, $R_{n-1,0}$ and $R_{n-1,1}$. Among these three points, the first two are available since $R_{i,0} = A(i)/4$. The third point is given by the expression $R_{n-1,1} = 2A_n/(8 + \epsilon^2 {A_n}^2)$, where $\epsilon = \frac{1}{n}$ is the discrete step size and n is the number of layers. With these initial values, the computation is favored in parallel implementation.

For the inverse problem, the left boundary is the homogeneous medium. Since the modification needs the knowledge of two layers to initiate the computation, we add an extra layer from the homogeneous medium. As seen from Fig. 2, since the homogeneous medium properties match that of the inhomogeneous medium at the common boundary, all we need to do is to replicate the leftmost layer. This initializes the inverse algorithm. The modified inverse algorithm is also well suited for parallel implementation.

The governing equations for the forward and the inverse problems are the same except for the initializations. An extra layer in the homogeneous region is used to initialize the inverse problem. However the additional layer is identical to the boundary layer between the homogeneous and the inhomogeneous medium. This result follows by the assumptions made in section 2. In fact, the success of the reconstruction using the results from the forward algorithm as the input to the inverse algorithm will be one of the strong indicators that the modifications and the initializations made in this paper are valid. Another way to verify the correctness of the modified equations is to solve the forward problem using the algorithm presented in section 2.2.1, and use the computed reflection kernel to reconstruct the medium using the modified inverse algorithm. These two cases were implemented as follows:

5.1.1 Case1: Computations using modified forward and inverse algorithms

- $A(x) = sin(4\pi x) + 2$
- Solve the forward algorithm using the modified forward algorithm.
- Use the knowledge of the computed reflection kernel and the modified inverse algorithm to reconstruct $A(x)$.

Input and the reconstructed $A(x)$ are shown in Fig. 6.

5.1.2 Case2: Computations using conventional forward and modified inverse algorithms

- $A(x) = sin(4\pi x) + 2$
- Solve the forward algorithm using the conventional forward algorithm.
- Use the knowledge of the computed reflection kernel and the modified inverse algorithm to reconstruct $A(x)$.

Input and the reconstructed $A(x)$ are shown in Fig. 7.

In the second example, $A(x)$ is piece wise continuous function with

$$A(x) = \begin{cases} 5 & 0 < x < 10 \\ -5 & 10 \leq x < 16 \\ sin(4\pi x) + 5 & 16 \leq x < 50 \\ -5 & 50 \leq x < 64 \end{cases}$$

Forward and the inverse problem were solved using the modified algorithms in parallel implementation. The resulting reconstruction of $A(x)$ is plotted in Fig. 8.

Computational timings of the conventional forward and the modified parallel forward algorithms were measured for various discretization step sizes, and their ratio in log scale was ploted in Fig. 9. (The timing ratio is defined as the **log[conventional timing/modified timing]**.) Using a least square fit it was noted that the parallel version of the algorithm takes $\bigcirc(N^2)$ computations as expected. The parallelized and the sequential versions of the algorithms were implemented in MasPar using MPL C language.

6 Conclusion

In this paper, we presented a reformulation of a class of 1-D forward and inverse scattering problems, employing invariant imbedding methods. The new formulation decouples the reflection kernel in a given layer.

It was shown that the smoothness assumptions made in the formulation of the original problems were sufficient for the reformulation. Moreover, these assumptions also led to efficient initialization schemes for the parallel implementation of the inverse problem.

The error in the reformulation was shown to be controlled by the step size (varied as the square of the step size). Due to its decoupling nature, the new formulation completely eliminated any error propagation between any two points in the same layer. The error propagation was tractable for the forward and the inverse problems.

Numerical results showed the performance of the modified algorithm was indeed as expected from the analytical derivation presented in the paper.

Acknowledgements

We wish to thank Ms. Houra Rais of McDonnell Douglas for many useful suggestions to improve the quality of the presentation.

References

[1] J.P. Corones, M.E. Davison, and R.J. Krugger, "Direct and inverse scattering in the time domain via invariant imbedding equations," *J.Acoust.Soc.Am. 74(5)*, Nov. 1983, pp. 1535-1541.

[2] J.P. Corones, M.E. Davison, and R.J. Krugger, "Wave splittings, invariant imbedding and inverse scattering," *Inverse Optics, Proc. SPIE 413*, edited by Anthony J. Devaney (SPIE, Bellingham, WA, 1983), pp. 102-106.

[3] J.P. Corones, M.E. Davison, and R.J. Krugger, "The effect of dissipation in one-dimensional inverse problems," *Inverse Optics, Proc. SPIE 413*, edited by Anthony J. Devaney (SPIE, Bellingham, WA, 1983), pp. 107-114.

[4] A. E. Yagle and B. C. Levy,"Application of Schur algorithm to the inverse problem for a layered acoustic medium," *J.Acoust.Soc.Am. 76(1)*. July 1984, pp. 301-308.

[5] P. Raadhakrishnan,"A new parallel algorithm for a class of direct and inverse Scattering Problems," *Internal Quarterly Report, USRA*, July-Sept. 1991.

[6] P. Raadhakrishnan and J. E. Dorband, "An algorithm for a class of direct and inverse scattering problems," *Fourth Symposium on the Frontiers of Massively Parallel Computation, IEEE Computer Society* , October 19-21, 1992, pp. 237-243.

[7] A. E. Yagle and P. Raadhakrishnan,"Numerical performance of the layer stripping algorithms for two-dimensional inverse scattering problems," *Inverse Problems 8* 1992 pp. 645-665.

[8] A. M. Bruckstein, B. C. Levy and T. Kailath, "Differential methods in inverse scattering," *SIAM J.APPL. MATH*, Vol. 45, 2, April 1985, pp. 312-335.

List of Figures

1 Velocity profile c(z) as a function of the medium depth. 14

2 Computational grid for the forward problem 14

3 Computational grid for the inverse problem 15

4 Modified computational grid for the forward problem. 15

5 Modified computational grid for the inverse problem. 16

6 Computations using modified forward and inverse algorithms. 16

7 Computations using conventional forward and modified inverse algorithms. 17

8 Computations using modified forward and inverse algorithms. 17

9 Computational timing ratio of the conventional forward algorithm and the modified parallel forward algorithm in log scale as a function of the discretization points. . . . 18

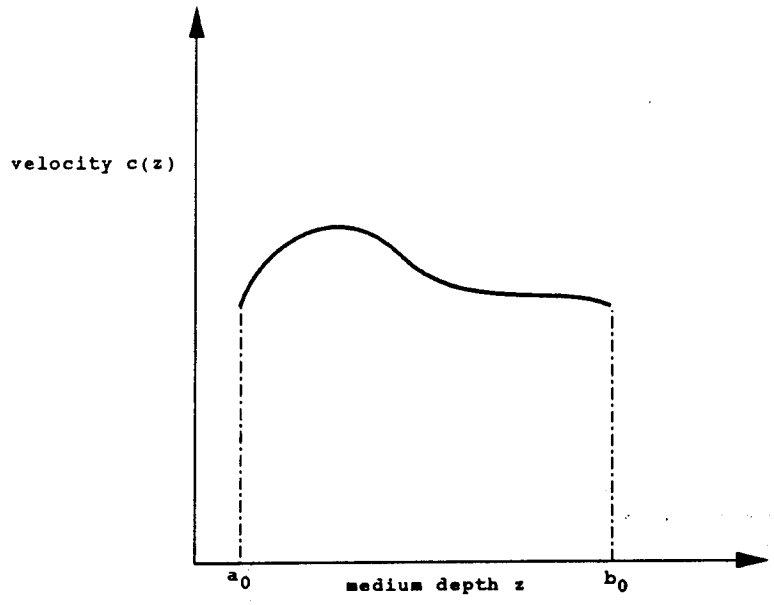


Figure 1: Velocity profile c(z) as a function of the medium depth.

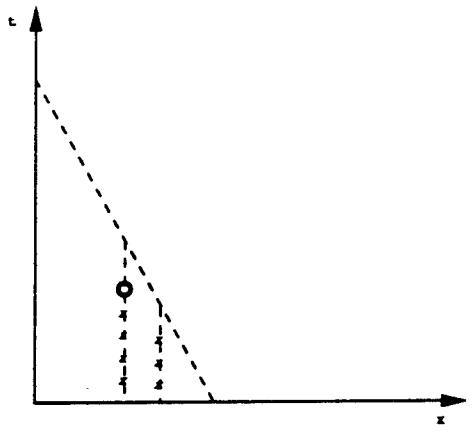


Figure 2: Computational grid for the forward problem

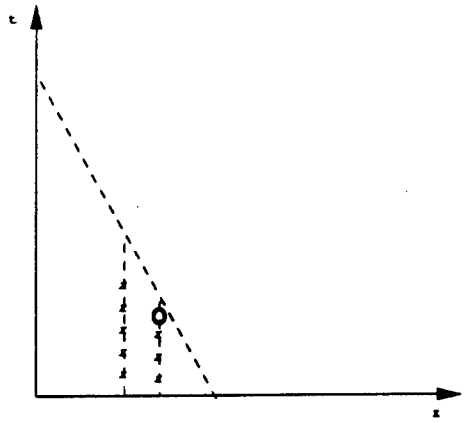


Figure 3: Computational grid for the inverse problem

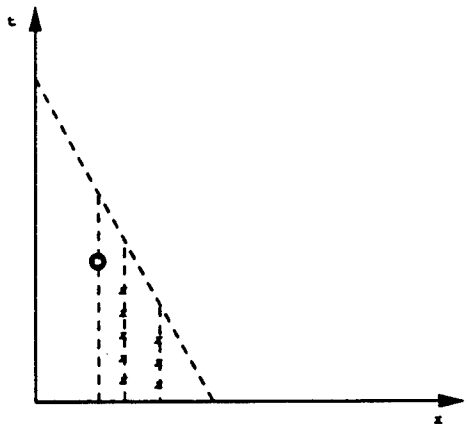


Figure 4: Modified computational grid for the forward problem.

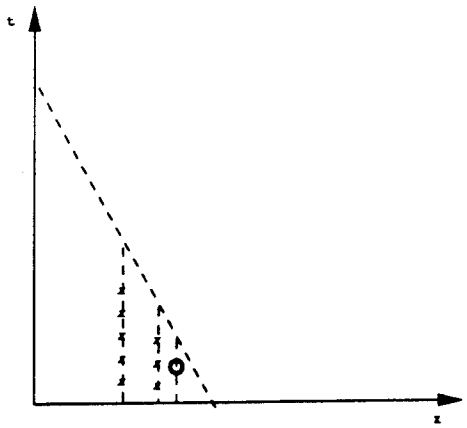


Figure 5: Modified computational grid for the inverse problem.

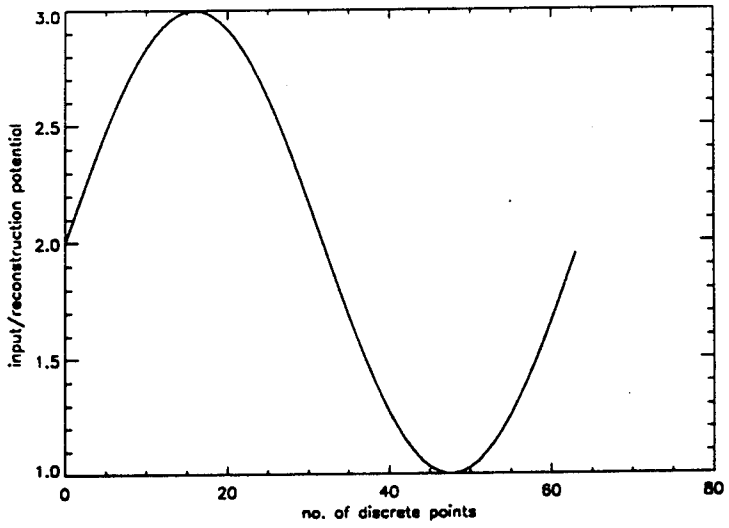


Figure 6: Computations using modified forward and inverse algorithms.

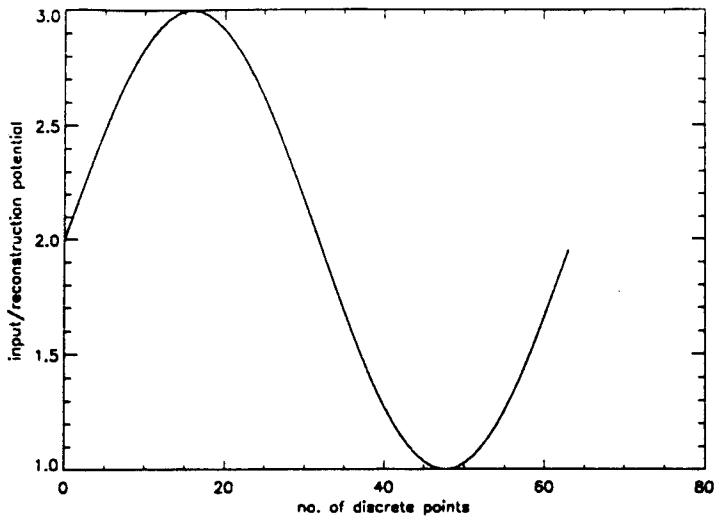


Figure 7: Computations using conventional forward and modified inverse algorithms.

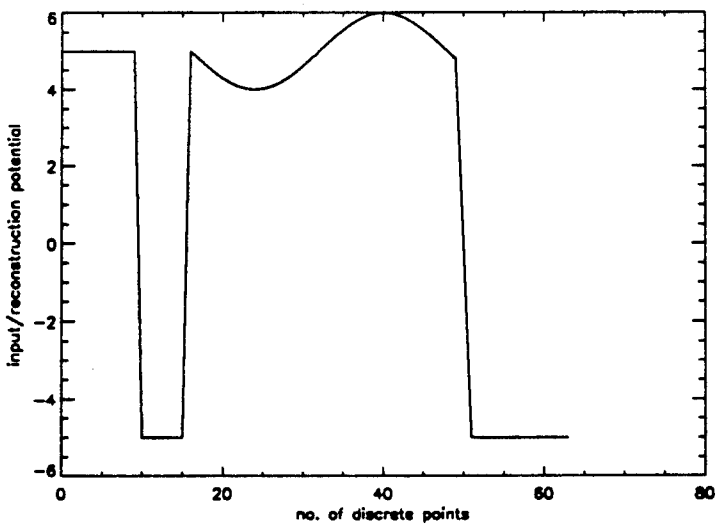


Figure 8: Computations using modified forward and inverse algorithms.

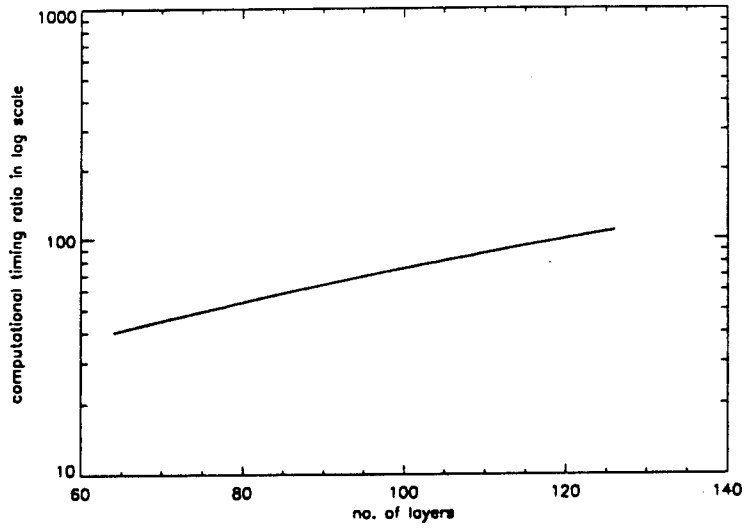


Figure 9: Computational timing ratio of the conventional forward algorithm and the modified parallel forward algorithm in log scale as a function of the discretization points.