NPS-EC-99-004

NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIFORNIA

THESIS

INVESTIGATION OF NEAR-FIELD ELECTROMAGNETIC SOURCE IMAGING USING INVERSE GREEN'S FUNCTION INTEGRATIONS

by

Daryl G. Steenman

March 1999

Thesis Advisor: Michael A. Morgan

Approved for public release; distribution is unlimited.

Prepared for:
Office of Naval Research
800 N. Quincy Street
Arlington, VA 22217

NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIFORNIA 93943

Rear Admiral Robert C. Chaplin
Superintendent

This thesis was prepared in conjunction with research sponsored in part by the Office of Naval Research, 800 N. Quincy Street, Arlington, VA 22217.

Released by:

David W. Netzer, Associate Provost and Dean of Research

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE March 1999

3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE INVESTIGATION OF NEAR-FIELD ELECTROMAGNETIC SOURCE IMAGING USING INVERSE GREEN'S FUNCTION INTEGRATIONS

5. FUNDING NUMBERS N0001499WR20039

6. AUTHOR(S) Daryl G. Steenman in conjunction with Michael A. Morgan and David C. Jenn

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER NPS-EC-99-004

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Office of Naval Research
800 N. Quincy Street, Arlington, VA 22217

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

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

12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)*

As continued efforts are made to reduce the radar cross sections of aircraft and ships, designs are first modeled with computers and then tested in the lab. In the far-field of these tested objects, actual sources of high reflectivity or "Hot Spots" on the tested objects can be isolated to within only one half the wavelength of the electromagnetic wave used for testing. Ideally, a probe could measure fields on the surface of the object being tested to completely isolate the source of the hot spot. Unfortunately, the presence of the probe on the surface of the object will disturb the very fields it is attempting to measure. Probe measurements made in the near field, close to but not on the object, can be designed to reduce the influence of the probe while providing accurate field data. The data thus measured, while not able to determine the source location perfectly, can be used to localize a source to less than one half wavelength, the far-field diffraction limit. This thesis tests a technique for back propagating computer generated near field measurements of an axisymmetric field source to determine the fields closer to the source. Several cases are examined that test the accuracy and resolving capability of the technique.

SUBJECT TERMS Electromagnetic Waves, Electromagnetic Imaging, Inverse Source Problem, Back Propagation

15. NUMBER OF PAGES 115

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified

18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified

19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified

20. LIMITATION OF ABSTRACT UL

INVESTIGATION OF NEAR-FIELD ELECTROMAGNETIC SOURCE IMAGING USING INVERSE GREEN'S FUNCTION INTEGRATIONS

Daryl G. Steenman
Lieutenant, United States Navy
B.S., Rice University, 1991

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
March 1999

Author: Daryl G Steen
Daryl G. Steenman

Approved by: Michael A. Morgan
Michael A. Morgan, Thesis Advisor

David C. Jenn
David C. Jenn, Second Reader

Jeffrey B. Knorr, Chairman
Department of Electrical and Computer Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

As continued efforts are made to reduce the radar cross sections of aircraft and ships, designs are first modeled with computers and then tested in the lab. In the far-field of these tested objects, actual sources of high reflectivity or “Hot Spots” on the tested objects can be isolated to within only one half the wavelength of the electromagnetic wave used for testing. Ideally, a probe could measure fields on the surface of the object being tested to completely isolate the source of the hot spot. Unfortunately, the presence of the probe on the surface of the object will disturb the very fields it is attempting to measure. Probe measurements made in the near field, close to but not on the object, can be designed to reduce the influence of the probe while providing accurate field data. The data thus measured, while not able to determine the source location perfectly, can be used to localize a source to less than one half wavelength, the far-field diffraction limit. This thesis tests a technique for back propagating computer generated near field measurements of an axisymmetric field source to determine the fields closer to the source. Several cases are examined that test the accuracy and resolving capability of the technique.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGEMENT

The professional guidance and considerable knowledge provided by my advisor, Dr. Michael A. Morgan, made this study possible.

I would also like to thank my wife. Without her support, time, patience and flexibility I would not have been able to finish this thesis.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

The harder it is to detect a platform, the easier it is for a platform to escape observation and targeting. This is the driving force behind research and engineering in the area of "stealth." Stealth refers not only to the oddly shaped airframe of the F-117, but to a wide realm of techniques designed to reduce the many facets of a platform's "signature." The signature of a platform consists of acoustic, thermal, and electromagnetic sources that are exploited in the detection of that platform. As stealth technology has become more publicized, various efforts for countering it have also been under exploration. Advances made in overcoming stealth directly imply that continued efforts are required to further reduce signatures.

A. GOAL OF RESEARCH

In the areas of Radar Cross Section (RCS) and electromagnetic sources, the general trend has been to attempt back-propagation of field measurements from the far field region of a platform down to the surface of the platform itself. From this estimation of the fields on the platform, the distribution, localization, and characterization of source currents can be approximated. Unfortunately, sources can only be distinguished separately if they are greater than one half wavelength apart when measurements are taken from the far field. Figure 1 shows the relationship of three regions of interest to the surface of a source and the regions' resolution limitations.

Alternatively, since exact source localization would be ideal, field measurements made directly on the platform by a perfect probe might yield the required information. Unfortunately, the presence of an actual probe will greatly disturb the fields and source distribution on the platform.

This leaves the near field region of the platform, between the platform's surface and the far field, in which to make measurements and to image the sources. For a probe in the near field, the measurable fields no longer constrain the source resolution to structures greater than one half wavelength (as was the case for the far field) and allow for greater refinement. Additionally, the probe can be placed at a distance sufficiently far away from the surface to make the effects of its presence managable.

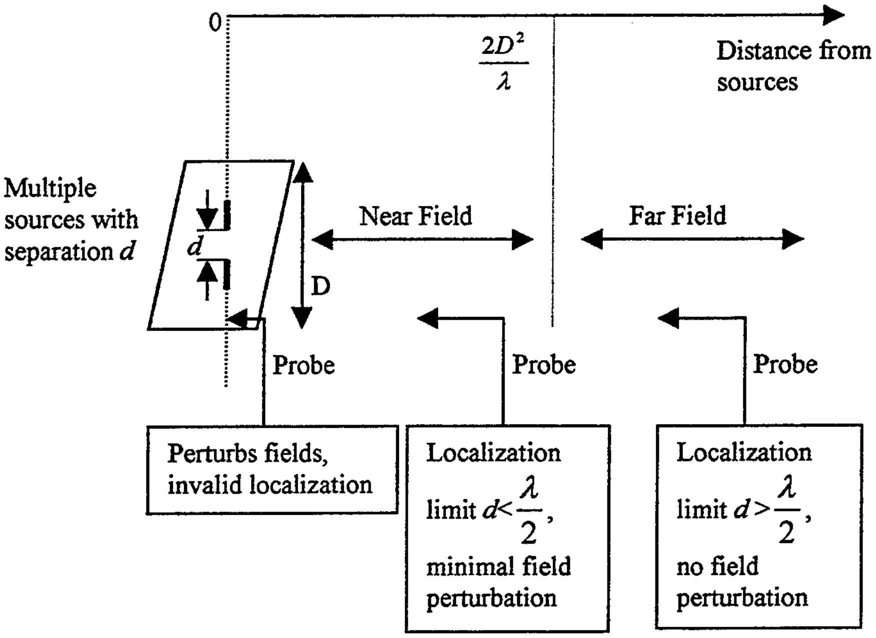


Figure 1. Far Field/Near Field/Surface Localization Limitations

Exploring a technique for the back propagation of fields measured in the near field region at one distance from the source onto fields closer to the source is the primary goal of this thesis.

B. BACKGROUND OF RESEARCH

The analysis of near field phenomena that spurred the current research originated in the field of acoustics. For plate and cylinder like structures, a newly defined quantity called the supersonic acoustic intensity vector provided a tool for the location and characterization of acoustic sources from near field acoustic measurements. [Ref. 1] From acoustics, the supersonic analysis of sound wave components was translated into the realm of electromagnetics. For cylindrical geometry, the now superluminal modes which satisfy $|k_z| < k$ and have faster-than-light propagation along the cylindrical axis provide time-average power to the far field. Measurements taken in the near field were back-propagated using an optimal deconvolution filter to account for the subluminal modes. [Ref. 2] This technique of using superluminal modes was then extended from separable geometries to non-separable geometries. Using the Finite Element Method to relate the boundary conditions of a measured field to the surface currents induced on an axisymmetric body by scattering, fields were back-propagated to estimate the source distribution. Resonance issues in the spherical test case complicated the overall effectiveness of that particular approach. [Ref. 3]

This thesis continues to explore methods of near field back-propagation that apply to general non-separable geometries from axisymmetric bodies. These are bodies that do not have explicit separable modes such as cylindrical or spherical wave functions. Unlike the previous work in non-separable geometries, this thesis attempts to reconstruct fields closer to the source and not strictly the source itself. Additionally, this thesis looks at source current distributions that are not induced by scattering.

This study is organized as follows: Various cases of generated electromagnetic radiation source fields are discussed in Chapter II. Various techniques in calculating forward propagation fields and testing their convergence are examined in Chapter III. Various techniques in calculating back propagation of fields are discussed in Chapter IV. A summary and concluding discussion are presented in Chapter V. An appendix has been included with the fundamental set of computer codes to allow for an indepth review and continuation of this line of research.

THIS PAGE INTENTIONALLY LEFT BLANK

II. GENERATION OF ELECTROMAGNETIC SOURCE FIELDS AND THE FIELD TRANSFORMATION OPERATOR

A. GENERATION OF THE EXACT FIELDS

Before one can perform numerical experiments on the accuracy of back-propagated fields, one must generate the fields to be propagated and transform matrix that does the propagation operations. First we consider a single dipole located at the origin. Following the derivation [Ref. 4], a closed form solution to the radiation patterns any distance away from the source exists. These closed form solutions form the exact fields at various ρ values which are in turn used for two purposes. First, the exact fields are used as checks of the accuracy for various numerical techniques, and second, as the basis for determining the equivalent electric and magnetic currents required for numerical propagation. The dipole is placed at the origin with its length along the z-axis as shown in Figure 2.

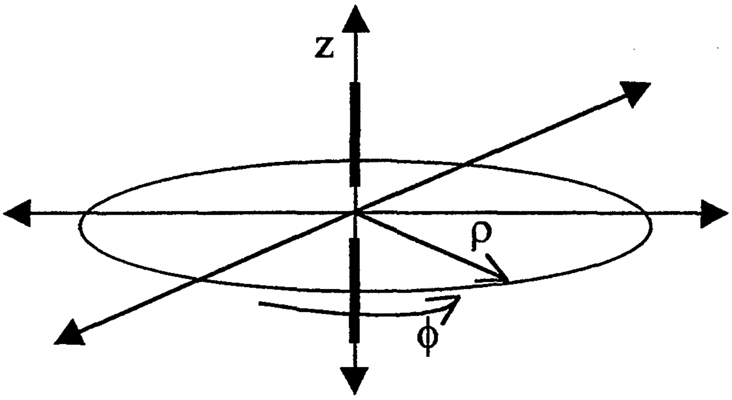


Figure 2. Electric Dipole with Cylindrical Coordinates

The axisymmetric fields it produces will be the same at any point with the same ρ and z around the axis. Any combination of dipoles oriented along the z-axis will also produce axisymmetric fields. The closed form solutions are based on an analysis of a dipole with a sinusoidal current distribution. Figure 3 shows the dipole geometry used in the derivation. The current distribution on the dipole is sinusoidal such that

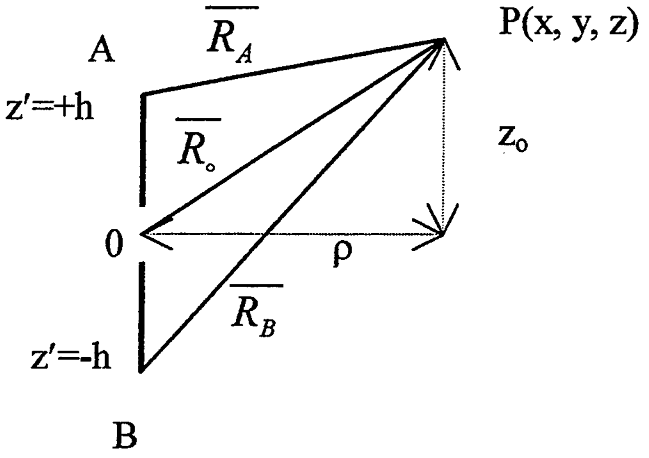
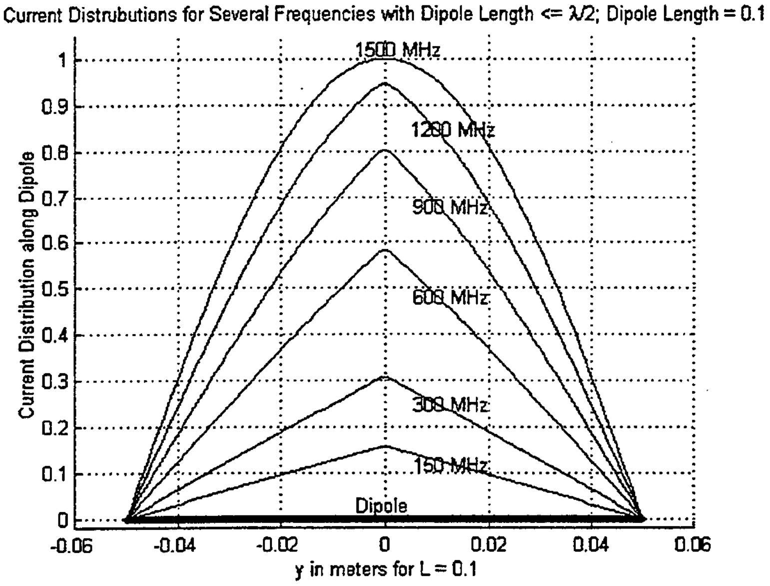


Figure 3. Dipole Exact Field Derivation Geometry

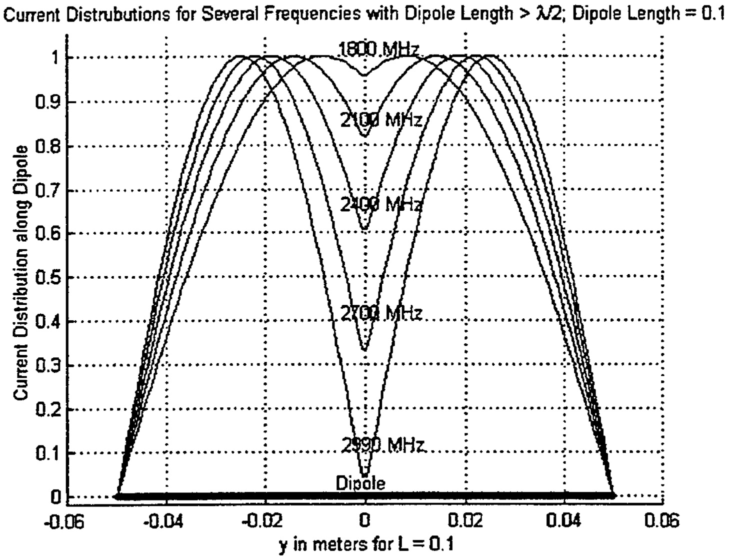
$I_e = I_{DIPOLE} \sin k(h - |z'|)$, where $z' \in [+h, -h]$. Figure 4 illustrates two cases. The first case shows the dipole with an overall length less then one half the wavelength associated with the current frequency. The second case shows the dipole with a length between one half and one wavelength. Phasor Fields may be determined through use of the vector potential, computed for line currents as [Ref. 5]

$$\overline{A}(x,y,z) = \frac{\mu}{4\pi} \int_c \overline{I_e}(x',y',z') \frac{e^{-jkR}}{R} dz' \tag{1}$$

with $R = \sqrt{x^2 + y^2 + (z - z')^2}$. (2)



Current Distrubutions for Several Frequencies with Dipole Length <= λ/2; Dipole Length = 0.1



Current Distrubutions for Several Frequencies with Dipole Length > λ/2; Dipole Length = 0.1

Figure 4. Sinusodial Current Distribution

Shifting to cylindrical coordinates and substituting the sinusoidal current distribution of the dipole for $\overline{I_e}(x',y',z')$, the integration may be readily performed. Using Maxwell's equations, the magnetic field due to vector potential $\overline{A}(\rho,\phi,z)$ is given by $\overline{H_A}=\frac{1}{\mu_\circ}\left(\nabla\times\overline{A}\right)$. The resulting electric field is given by $\overline{E}=\frac{1}{j\omega\varepsilon_\circ}\left(\nabla\times\overline{H}\right)$. The fields produced by a sinusoidal current distribution can then be derived as the following,

$$H_\phi = -\frac{I_{DIPOLE}\,\mu_\circ}{j4\pi\rho}\left[e^{-jkR_A}+e^{-jkR_B}-(2\cos kh)e^{-jkR_\circ}\right] \tag{3}$$

$$E_\rho = j30I_{DIPOLE}\left[\left(\frac{z-h}{\rho}\right)\frac{e^{-jkR_A}}{R_A}+\left(\frac{z+h}{\rho}\right)\frac{e^{-jkR_B}}{R_B}-(2\cos kh)\frac{z}{\rho}\frac{e^{-jkR_\circ}}{R_\circ}\right] \tag{4}$$

$$E_z = -j30I_{DIPOLE}\left[\frac{e^{-jkR_A}}{R_A}+\frac{e^{-jkR_B}}{R_B}-(2\cos kh)\frac{e^{-jkR_\circ}}{R_\circ}\right] \tag{5}$$

These three equations are functions of

1) Dipole height, h,
2) Wavelength, $k=\frac{2\pi f}{c}$,
3) Maximum Current Magnitude, I_{DIPOLE}, and the
4) Observation point in ρ and z, but not ϕ since the fields are axisymmetric.

Using H_ϕ and E_z, the fields tangential to a cylindrical surface of any radius ρ can be determined within machine round off accuracy.

B. TANGENTIAL FIELD GENERATION FROM EXACT AXISYMMETRIC FIELDS ON A CYLINDRICAL SURFACE

The fields on a cylindrical surface at ρ_2 can be determined in two ways. First, the fields can be calculated from the actual source axisymmetric current distribution.

Second, if the fields are exactly known on some closed surface that contains the actual sources and is inside the ρ_2 cylindrical surface, then the surface equivalence theorem can be used to transform the known fields into electric and magnetic currents on the closed surface. Since the source current distribution is axisymmetric, the inner surface is chosen to be an infinite cylindrical surface. Figure 5 shows the relative geometry of the two coaxial cylindrical surfaces used in determining the fields at ρ_2.

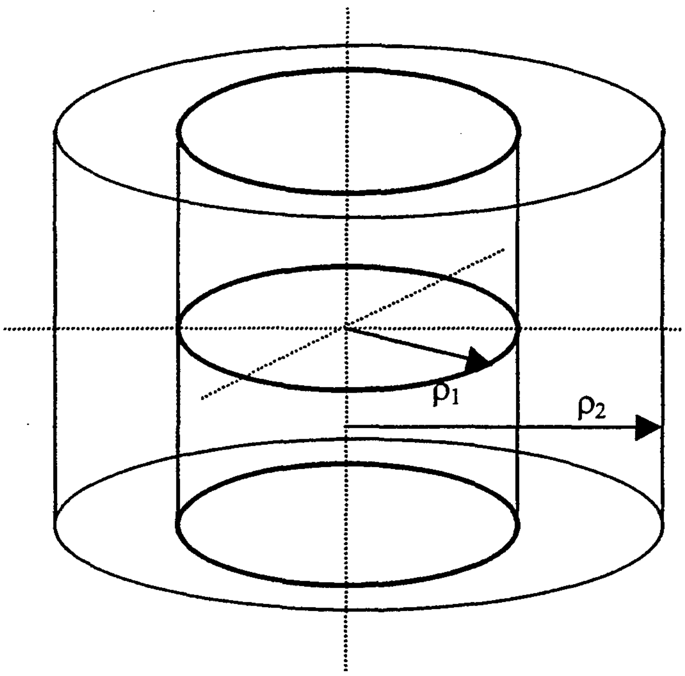


Figure 5. Coaxial Cylindrical Surfaces Geometry

The exact fields from the dipole source have both E- and H-field components tangential to a cylindrical surface as shown in equations (3) and (5). Applying the surface equivalence theorem, the fields at $\rho_{1,}$ become the following equivalent surface currents.

$$\overline{J}_{surface} = \hat{n}_1 \times \overline{H} = \hat{\rho}_1 \times H_\phi(\rho_1, \phi_1, z_1)\hat{\phi} = (H_\phi)\hat{z}_1 = J_z \hat{z}_1 \qquad (6)$$

$$\overline{M}_{surface} = \overline{E}\times\hat{n}_1 = E_z(\rho_1,\phi_1 z_1)\hat{z}\times\hat{\rho}_1 = E_z\hat{\phi}_1 = M_\phi\hat{\phi}_1 \qquad (7)$$

Using $\overline{J}_{surface}$ and $\overline{M}_{surface}$, the vector potentials $\overline{A}$ and $\overline{F}$ can be determined via the inhomogeneous vector potential equations.

$$\overline{A} = \left(\frac{\mu_\circ}{4\pi}\right)\iint_{s_1}\overline{J}_{surface}\frac{e^{-jkR}}{R}ds' \qquad (8)$$

$$\overline{F} = \frac{\varepsilon_\circ}{4\pi}\iint_{s_1}\overline{M}_{surface}\frac{e^{-jkR}}{R}ds' \qquad (9)$$

Substituting $\overline{A}$ and $\overline{F}$ into Maxwell's equations yields the following component fields [Ref.5].

$$\overline{H}_A = \left(\frac{1}{\mu_\circ}\right)\nabla\times\overline{A} \qquad (10)$$

$$\overline{E}_A = \left(\frac{1}{j\omega\varepsilon_\circ}\right)\nabla\times\overline{H}_A \qquad (11)$$

$$\overline{E}_F = -\frac{1}{\varepsilon_\circ}\nabla\times\overline{F} \qquad (12)$$

$$\overline{H}_F = -\frac{1}{j\omega\mu_\circ}\nabla x\overline{E}_F \qquad (13)$$

$$\overline{H} = \overline{H}_A + \overline{H}_F \qquad (14)$$

$$\overline{E} = \overline{E}_A + \overline{E}_F$$

In order to determine $\overline{H}$ and $\overline{E}$, ultimately the Green's kernel $\frac{e^{-jkR}}{R}$ will have to be integrated. The following subsection sets up the problem for that integration.

1. Setting Up the Problem

The geometry for an arbitrary field point is shown in Figure 6.

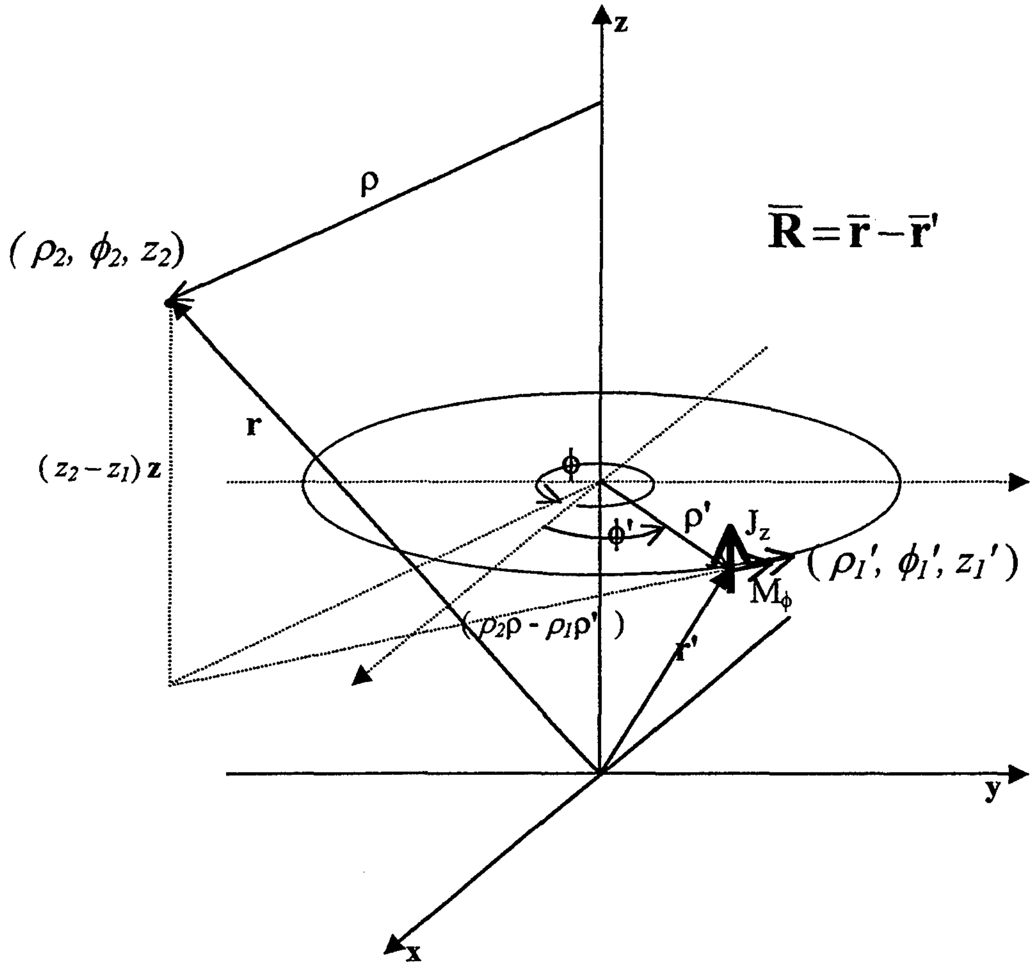


Figure 6. General Case for an Arbitrary Field Point (ρ_2, ϕ_2, z_2)

The steps for setting up the problem for an arbitrary field point are shown below.

$$\hat{\rho} = \cos\phi_2\hat{x} + \sin\phi_2\hat{y} \tag{15}$$

$$\hat{\phi} = \cos(90+\phi_2)\hat{x} + \sin(90+\phi_2)\hat{y} \tag{16}$$

$$\hat{\rho}' = \cos\phi_1\hat{x} + \sin\phi_1\hat{y} \tag{17}$$

$$\hat{\phi}' = \cos(90+\phi_1)\hat{x} + \sin(90+\phi_1)\hat{y} \tag{18}$$

$$\hat{z} = \hat{z}' \tag{19}$$

$$\overline{r'} = \rho_1 \hat{\rho}' + z_1 \hat{z}' \tag{20}$$

$$\overline{r} = \rho_2 \hat{\rho} + z_2 \hat{z} \tag{21}$$

$$\overline{R} = \overline{r} - \overline{r'} = \rho_2 \hat{\rho} - \rho_1 \hat{\rho}' + (z_2 - z_1)\hat{z} \tag{22}$$

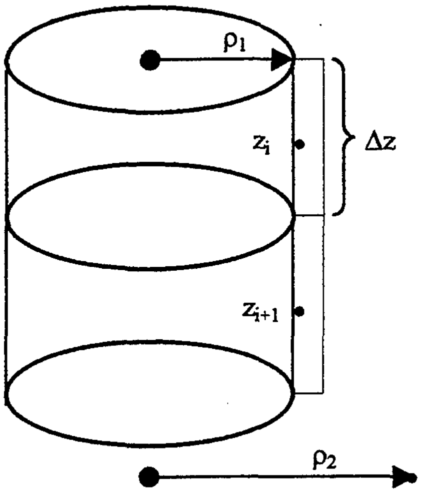
$$\overline{R} = [\rho_2 - \rho_1 \cos(\phi_1 - \phi_2)]\hat{\rho} + [\rho_1(-\sin(\phi_1 - \phi_2))]\hat{\phi} + (z_2 - z_1)\hat{z} \tag{23}$$

$$\overline{J} = J_z \hat{z}' = J_z \hat{z} \tag{24}$$

$$\overline{M} = M_\phi \hat{\phi}' = [M_\phi(-\sin(\phi_1 - \phi_2))]\hat{\rho} + [M_\phi(\cos(\phi_1 - \phi_2))]\hat{\phi} \tag{25}$$

$$R = \left|\overline{R}\right| = \sqrt{\rho_2^2 + \rho_1^2 - 2\rho_1\rho_2 \cos(\phi_1 - \phi_2) + (z_2 - z_1)^2} \tag{26}$$

There does not appear to be a simple closed form solution to the Green's kernel integration over the infinite cylindrical surface required to find $\overline{A}$ and $\overline{F}$. However, there is a straightforward numerical technique for solving the integration. First one divides the cylinder at ρ_1 into many small bands or rings that have a constant current magnitude on the ring as shown in Figure 7. Then the ring is subdivided into many smaller squares such that both electric and magnetic equivalent currents are constant on the square [Figure 8].



Δz corresponds to the width of the thin ring such that

$$z_i - \frac{\Delta z}{2} \le z_i \le z_i + \frac{\Delta z}{2}$$

Figure 7. Physical Meaning of Δz

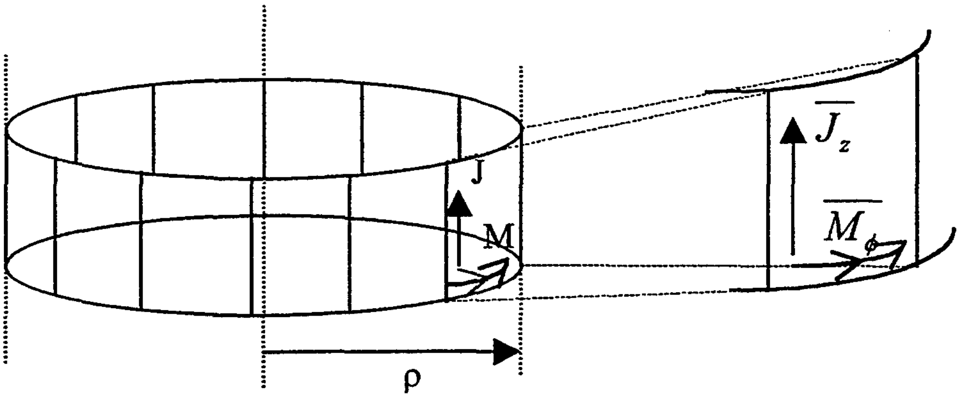


Figure 8. Subdivision of the Ring Showing Constant Equivalent Currents

The influence of the currents in each square on a ring to the field at ρ_2 will be summed together to give the total influence of the ring. Since the fields at all ρ are axially symmetric, the influence of whole rings can be used when setting up the mathematical (matrix) representation of the system, as opposed to the individual influences of each small square on each ring separately. One sums the respective contributions to an individual point on the ρ_2 cylindrical surface made by each ring along the length of the inner cylinder to get an estimate of the actual field at that point on ρ_2. This is repeated for each point on ρ_2 in order to obtain the estimated fields on the entire ρ_2 cylinder.

Taking advantage of the axial symmetry of the fields, the field point to be estimated is placed on the x-z plane, where y=0 and ϕ_2=0 [Figure 9].[Ref. 6] When the field point is on the x-z plane, the problem set up is simplified.

$$\hat{\rho} = \hat{x} \tag{27}$$

$$\hat{\phi} = \hat{y} \tag{28}$$

$$\hat{z} = \hat{z} \tag{29}$$

$$\overline{M} = \left(M_\phi \sin(-\phi_1)\right)\hat{\rho} + \left(M_\phi \cos\phi\right)\hat{\phi} \tag{30}$$

$$\overline{R} = \left[\rho_2 - \rho_1 \cos\phi_1\right]\hat{\rho} + \left[\rho_1(-\sin\phi_1)\right]\hat{\phi} + \left[z_2 - z_1\right]\hat{z} \tag{31}$$

$$R = \sqrt{\rho_2^2 + \rho_1^2 - 2\rho_1\rho_2 \cos\phi_1 + \left(z_2 - z_1\right)^2} \tag{32}$$

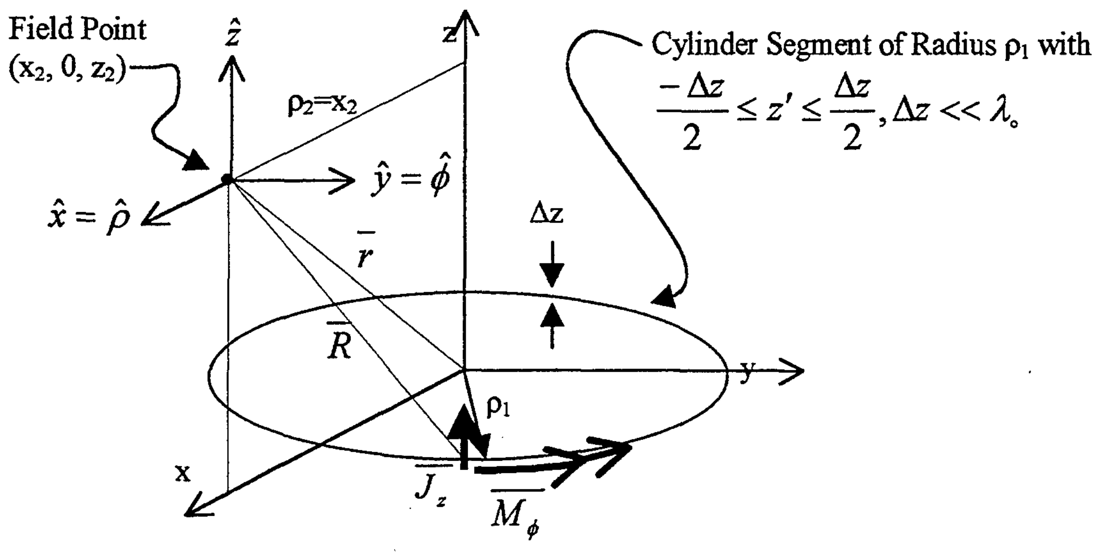


Figure 9. Field Point (x_2, 0, z_2) and the Cylinder Segment

By using a ring of width Δz_1, $\overline{A}$ is calculated by the summation of ΔA_z segments.

$$A_z\hat{z} = \frac{\mu_\circ}{4\pi}\left[\sum_{i=-\infty}^{\infty}\int_{-\pi}^{\pi}(J_z)_i\hat{z}\frac{e^{-jkR_i}}{R_i}\rho_1 d\phi_1 \Delta z\right] \tag{33}$$

Therefore, the i^{th} segment of $A_z=(\Delta A_z)_I$ is as follows.

$$(\Delta A_z)_i = \frac{\mu_\circ \rho_1 \Delta z}{4\pi}\int_{-\pi}^{\pi}(J_z)_i\frac{e^{-jkR_L}}{R_i}d\phi_1 \tag{34}$$

In the limit of Δz → dz, ΔA_z → dA_z, and $\sum_{i=-\infty}^{\infty}(\Delta A_z)_i \to \int_{-\infty}^{\infty} dA_z = A_z$.

$$\Delta A_z = \frac{\mu_\circ \rho_1 \Delta z}{4\pi}\int_{-\pi}^{\pi}J_z\frac{e^{-jkR}}{R}d\phi_1 \tag{35}$$

Following a similar set up $\overline{F}$ and $\overline{\Delta F}$ are found to be the following.

$$\overline{F} = \frac{\varepsilon_\circ}{4\pi}\iint_s \overline{M_s}(r')\frac{e^{-jkR}}{R}ds' = \frac{\varepsilon_\circ}{4\pi}\sum_{i=-\infty}^{\infty}\int_{-\pi}^{\pi}\left(\overline{M_s}(r')\right)_i\frac{e^{-jkR_i}}{R_i}\rho_1 d\phi_1 \Delta z \tag{36}$$

$$\overline{\Delta F} = \frac{\varepsilon_o \rho_1 \Delta z}{4\pi} \int_{-\pi}^{\pi} \overline{M_s} \frac{e^{-jkR}}{R} d\phi_1 \qquad (37)$$

Once the vector potentials have been broken into segments for integration, the determination for the electric and magnetic field segments to be summed over the inner and outer cylinders is the next set of steps.

2. The Equivalent Electric Current Case and the Vector $\overline{A}$ Potential

Using ΔA_z as found in equation (35), $\overline{\Delta H_A}$ can be determined by the following equation.

$$\overline{\Delta \mathrm{H}_A} = \frac{1}{\mu_o} \nabla \times \overline{\Delta A} = \frac{1}{\mu_o} \nabla \times \left(\frac{\mu_o \rho_1 \Delta z}{4\pi} \int_{-\pi}^{\pi} (J_z \hat{z}) \frac{e^{-jkR}}{R} d\phi_1 \right) \qquad (38)$$

Since the curl operation is with regard to the coordinate system of the observer (ρ_2, ϕ_2, z_2) and the integration is with respect to ϕ_1, then the curl can be brought inside the integration as shown below.

$$\overline{\Delta \mathrm{H}_A} = \frac{\rho_1 \Delta z}{4\pi} \int_{-\pi}^{\pi} \left[\nabla \times \left((J_z \hat{z}) \frac{e^{-jkR}}{R} \right) \right] d\phi_1 =$$

$$\frac{\rho_1 \Delta z}{4\pi} \int_{-\pi}^{\pi} \left\{ \left[\left(\nabla \frac{e^{-jkR}}{R} \right) \times J_z \hat{z} \right] + \left[\frac{e^{-jkR}}{R} (\nabla \times J_z \hat{z}) \right] \right\} d\phi_1 \qquad (39)$$

Since $J_z \hat{z}$ is a function dependent only on ρ_1 and z_1 and the curl operation is with regard to the observer's coordinate system (ρ_2, ϕ_o, z_2), then $\nabla \times J_z \hat{z} = 0$. One will encounter many varieties of the operation shown below. [Ref. 6]

$$\nabla \frac{e^{-jkR}}{R^n} = \left(\frac{-n}{R^{n+2}} + \frac{-jk}{R^{n+1}} \right) e^{-jkR} \overline{R} \qquad (40)$$

This current case yields the following for n=1, $\nabla \frac{e^{-jkR}}{R} = \left(\frac{-1}{R^3} + \frac{-jk}{R^2} \right) e^{-jkR} \overline{R}$.

$$\overline{\Delta H_A} = \frac{\rho_1 \Delta z}{4\pi} \int_{-\pi}^{\pi} \left(\frac{-1}{R^3} + \frac{-jk}{R^2} \right) e^{-jkR} \left(\overline{R} \times J_z \hat{z} \right) d\phi_1 \tag{41}$$

$$\overline{R} \times J_z \hat{z} = J_z \left(\overline{R} \times \hat{z} \right) = J_z \left[R_\phi \hat{\rho} - R_\rho \hat{\phi} \right] \tag{42}$$

$$\overline{\Delta H_A} = \frac{\rho_1 \Delta z J_z}{4\pi} \int_{-\pi}^{\pi} \left(\frac{-1}{R^3} + \frac{-jk}{R^2} \right) e^{-jkR} \left((\rho_1 \cos\phi_1 - \rho_2) \hat{\phi} - (\rho_1 \sin\phi_1) \hat{\rho} \ \right) d\phi_1 \tag{43}$$

Therefore, one can say the following.

$$(\Delta H_A)_\phi = \frac{\rho_1 \Delta z J_z}{4\pi} \left[\rho_2 \int_{-\pi}^{\pi} \left(\frac{1}{R^3} + \frac{jk}{R^2} \right) e^{-jkR} d\phi_1 - \rho_1 \int_{-\pi}^{\pi} \left(\frac{1}{R^3} + \frac{jk}{R^2} \right) \cos\phi_1 e^{-jkR} d\phi \ \right] \tag{44}$$

$$(\Delta H_A)_\rho = \frac{\rho_1 \Delta z J_z}{4\pi} \left[\ \rho_1 \int_{-\pi}^{\pi} \left(\frac{1}{R^3} + \frac{jk}{R^2} \right) \sin\phi_1 e^{-jkR} d\phi_1 \ \right] \tag{45}$$

Since $R = \sqrt{\rho_2^2 + \rho_1^2 - 2\rho_1\rho_2 \cos\phi_1 + (z_2 - z_1)^2}$, it is an even function in ϕ_1 over the range $-\pi$ to $+\pi$. All functions of only R are even. All functions of R and $\cos\phi_1$ are also even. But, since $\sin\phi_1$ is an odd function, even functions of R and odd functions of $\sin\phi_1$ will be odd over the interval and will integrate to zero. Therefore, $(\Delta H_A)_\rho = 0$. Because one will encounter various versions of the functions of R and combinations of functions of R and $\cos\phi_1$ that are to be integrated, one defines U_n and V_n as well as W_n (which will be used in the next section), as follows. [Ref. 6]

$$U_n = \int_{-\pi}^{\pi} \frac{e^{-jkR}}{R^n} d\phi_1 \tag{46}$$

$$V_n = \int_{-\pi}^{\pi} \cos\phi_1 \frac{e^{-jkR}}{R^n} d\phi_1 \tag{47}$$

$$W_n = \int_{-\pi}^{\pi} (\ \cos\phi_1 \ )^2 \frac{e^{-jkR}}{R^n} d\phi_1 \tag{48}$$

U_n, V_n, and W_n are solved numerically by dividing the ring into many segments [Figure 5] and summing their individual components. The Matlab programs INTPHI*.M does this numerical integration step. Since J_z and M_ϕ are uniform around the ring and the differences arise from the physical distance to each segment and $\cos\phi_1$, then the segments can be summed around the ring. The ring will be the basic element for determining fields as opposed to each $\Delta\phi_1 x \Delta z$ segment. Overall, this reduces the computation time and simplifies the problem set up.

Substituting U_n, V_n, and W_n with applicable values for n yields the following.

$$(\Delta H_A)_\phi = \frac{\rho_1 \Delta z J_z}{4\pi}\left[\rho_2(U_3 + jkU_2) - \rho_1(V_3 + jkV_2)\right] \tag{49}$$

3. The Equivalent Magnetic Current Case and Vector $\overline{F}$ Potential

The mathematical details of the equivalent magnetic current case for H_A and the equivalent electric current for E_F are very similar in form.

$$\overline{\Delta E_F} = -\frac{1}{\varepsilon_o}\nabla \times \overline{\Delta F} \tag{50}$$

In $\overline{\Delta A}$, there was only a $\hat{z}$ component in $\overline{J_s}$ that contributed to $\overline{\Delta H_A}$. In $\overline{\Delta F}$ there are two components of $\overline{M_s}$: a $\hat{\rho}$ component and a $\hat{\phi}$ component. These two components add some complexity to the previous mathematical derivation of $\overline{\Delta H_A}$.

$$\overline{\Delta E_F} = -\frac{\rho_1 \Delta z}{4\pi}\int_{-\pi}^{\pi}\left[\nabla \times \left(\overline{M_s}\left(\frac{e^{-jkR}}{R}\right)\right)\right] d\phi_1 \tag{51}$$

$$\overline{\Delta E_F} = -\frac{\rho_1 \Delta z}{4\pi}\int_{-\pi}^{\pi}\left[\left(\nabla\frac{e^{-jkR}}{R}\right) \times \overline{M_s}\right] d\phi_1 \tag{52}$$

$$\overline{\Delta E_F} = -\frac{\rho_1 \Delta z}{4\pi}\int_{-\pi}^{\pi}\left(\frac{-1}{R^3} + \frac{-jk}{R^2}\right)\frac{e^{-jkR}}{R}\left(\overline{R} \times \overline{M_s}\right) d\phi_1 \tag{53}$$

Looking at the $\overline{R} \times \overline{M_s}$ term gives the following result.

$$\overline{R} \times \overline{M_s} = M_\phi\left[-(z_2 - z_1)(\cos\phi_1)\hat{\rho} - (z_2 - z_1)(\sin\phi_1)\hat{\phi} + (\rho_2\cos\phi_1 - \rho_1)\hat{z}\right] \tag{54}$$

Substituting $\overline{R} \times \overline{M_s}$ into $\overline{\Delta E_F}$ and breaking it into components yields the following result. [Ref. 6]

$$\left(\Delta E_F\right)_\rho = -\frac{M_\phi \rho_1 \Delta z}{4\pi}\left[\left(z_2 - z_1\right)\int_{-\pi}^{\pi}\left(\frac{1}{R^3} + \frac{jk}{R^2}\right)\cos\phi_1 e^{-jkR} d\phi_1\right] \tag{55}$$

$$\left(\Delta E_F\right)_\rho = -\frac{M_\phi \rho_1 \Delta z\left(z_2 - z_1\right)}{4\pi}\left[V_3 + jkV_2\right] \tag{56}$$

$$\begin{aligned}\left(\Delta E_F\right)_z = \frac{M_\phi \rho_1 \Delta z}{4\pi}\Bigg[\, &\rho_2 \int_{-\pi}^{\pi}\left(\frac{1}{R^3} + \frac{jk}{R^2}\right)\cos\phi_1 e^{-jkR} d\phi_1 - \\ &\rho_1 \int_{-\pi}^{\pi}\left(\frac{1}{R^3} + \frac{jk}{R^2}\right)_1 e^{-jkR} d\phi_1 \Bigg]\end{aligned} \tag{57}$$

$$\left(\Delta E_F\right)_z = \frac{M_\phi \rho_1 \Delta z}{4\pi}\left[\rho_2\left(V_3 + jkV_2\right) - \rho_1\left(U_3 + jkU_2\right)\right] \tag{58}$$

$$\left(\Delta E_F\right)_\phi = -\frac{M_\phi \rho_1 \Delta z}{4\pi}\left[\left(z_2 - z_1\right)\int_{-\pi}^{\pi}\left(\frac{1}{R^3} + \frac{jk}{R^2}\right)\sin\phi_1 e^{-jkR} d\phi_1\right] = 0 \tag{59}$$

4. Cross Field Influence Term

In Section II.B.2 "The Equivalent Electric Current Case and the Vector $\overline{A}$ Potential" set fourth above, $\overline{\Delta \mathrm{H}_A}$ at ρ_2 was calculated from $\overline{\Delta A}$, which was calculated from $\overline{J}$ which, in turn, was determined from a knowledge of the H-field at ρ_1. Therefore, $\overline{\Delta \mathrm{H}_A}$ is the part of the total $\overline{\Delta H}$ at ρ_2 determined strictly from the H-field at ρ_1. Similarly in Section III.B.3 "The Equivalent Magnetic Current Case and Vector $\overline{F}$ Potential," $\overline{\Delta E_F}$ at ρ_2 was calculated from $\overline{\Delta F}$, which was calculated from $\overline{M}$, which was determined from knowledge of the E-field at ρ_1. Therefore, $\overline{\Delta E_F}$ is the part of the total $\overline{\Delta E}$ at ρ_2 determined strictly from the E-field at ρ_1. But these are only parts of their respective total fields. Since $\overline{\Delta H}_{TOTAL} = \overline{\Delta H_A} + \overline{\Delta H_F}$ and $\overline{\Delta E}_{TOTAL} = \overline{\Delta E_F} + \overline{\Delta E_A}$, the influence on ΔH from the E-field at ρ_1 and the influence on ΔE from the H-field at ρ_1

must be determined in order to accurately predict the fields at ρ_2. Three ways to determine these field components were calculated.

a. *Method A: Maxwell's Equations Applied Directly to Calculated $\overline{\Delta \mathrm{H}_A}$ and $\overline{\Delta E_F}$ Fields*

This method, the most straightforward, takes the three previously calculated equations of $\overline{\Delta \mathrm{H}_A}$ and $\overline{\Delta E_F}$ and computes $\overline{\Delta E_A}$ and $\overline{\Delta \mathrm{H}_F}$ from them. Using Maxwell's equations and substituting the calculated $\overline{\Delta \mathrm{H}_A}$, one can state the following.

$$\overline{\Delta \mathrm{E}_A} = \frac{1}{j\omega\varepsilon_\circ}\nabla \times \overline{\Delta \mathrm{H}} = \frac{1}{j\omega\varepsilon_\circ}\nabla \times \Delta \mathrm{H}_\phi \hat{\phi} \tag{60}$$

$$\overline{\Delta \mathrm{E}_A} = \frac{1}{j\omega\varepsilon_\circ}\left\{-\frac{\partial \Delta \mathrm{H}_\phi}{\partial z_2}\hat{\rho} + \frac{1}{\rho_2}\frac{\partial}{\partial \rho_2}\left(\rho_2 \Delta \mathrm{H}_\phi\right)\hat{z}\right\} \tag{61}$$

$$\left(\Delta \mathrm{E}_A\right)_\rho = -\frac{\rho_1 \Delta z J_z}{j\omega\varepsilon_\circ 4\pi}\left\{\rho_2\left(\frac{\partial U_3}{\partial z_2} + jk\frac{\partial U_2}{\partial z_2}\right) - \rho_1\left(\frac{\partial V_3}{\partial z_2} + jk\frac{\partial V_2}{\partial z_2}\right)\right\} \tag{62}$$

$$\left(\Delta \mathrm{E}_A\right)_z = \frac{\rho_1 \Delta z J_z}{j\omega\varepsilon_\circ 4\pi}\left\{2\left[U_3 + jkU_2\right] - \frac{\rho_1}{\rho_2}\left[V_3 + jkV_2\right] + \right.$$

$$\left. \rho_2\left[\frac{\partial U_3}{\partial \rho_2} + jk\frac{\partial U_2}{\partial \rho_2}\right] - \rho_1\left[\frac{\partial V_3}{\partial \rho_2} + jk\frac{\partial V_2}{\partial \rho_2}\right]\right\} \tag{63}$$

Since all the representations contain partial derivatives of U_n and V_n with respect to ρ_2 and z_2, it is useful to calculate them in terms of other U_n, V_n, and W_n functions as shown below. [Ref. 6]

$$\frac{\partial}{\partial \rho_2}U_n = \rho_1\left(nV_{n+2} + jkV_{n+1}\right) - \rho_2\left(nU_{n+2} + jkU_{n+1}\right) \tag{64}$$

$$\frac{\partial}{\partial \rho_2}V_n = \rho_1\left(nW_{n+2} + jkW_{n+1}\right) - \rho_2\left(nV_{n+2} + jkV_{n+1}\right) \tag{65}$$

$$\frac{\partial}{\partial z_2}U_n = -(z_2 - z_1)(nU_{n+2} + jkU_{n+1}) \qquad (66)$$

$$\frac{\partial}{\partial z_2}V_n = -(z_2 - z_1)(nV_{n+2} + jkV_{n+1}) \qquad (67)$$

Substituting the partial derivatives of U_n and V_n into $\overline{\Delta E_A}$ and simplifying yields the following results.

$$(\Delta E_A)_\rho = \frac{\rho_1 \Delta z J_z (z_2 - z_1)}{j\omega\varepsilon_\circ 4\pi}\{\rho_2[3U_5 + 3jkU_4 - k^2 U_3]$$

$$- \rho_1[3V_5 + 3jkV_4 - k^2 V_3]\} \qquad (68)$$

$$(\Delta E_A)_z = \frac{\rho_1 \Delta z J_z}{j\omega\varepsilon_\circ 4\pi}\{2[U_3 + jkU_2] - \frac{\rho_1}{\rho_2}[V_3 + jkV_2]$$

$$+ \rho_2^2[k^2 U_3 - 3jkU_4 - 3U_5]$$

$$+ \rho_1 \rho_2[6V_5 + 6jkV_4 - 2k^2 V_3]$$

$$+ \rho_1^2[k^2 W_3 - 3jkW_4 - 3W_5]\} \qquad (69)$$

For the $\overline{\Delta H_F}$ fields the following can be stated.

$$\overline{\Delta H_F} = -\frac{1}{j\omega\mu_\circ}\nabla \times \overline{\Delta E_F} = -\frac{1}{j\omega\mu_\circ}\left\{\frac{\partial \Delta E_\rho}{\partial z_2} - \frac{\partial \Delta E_z}{\partial \rho_2}\right\}\hat{\phi} \qquad (70)$$

$$(\Delta H_F)_\phi = \frac{M_\phi \rho_1 \Delta z}{j\omega\mu_\circ 4\pi}\left\{\frac{\partial}{\partial z_2}[(z_2 - z_1)(V_3 + jkV_2)]\right.$$

$$\left. + \frac{\partial}{\partial \rho_2}[\rho_2(V_3 + jkV_2) - \rho_1(U_3 + jkU_2)]\right\} \qquad (71)$$

Substituting the partial derivatives of U_n and V_n into $\overline{\Delta H_F}$ and simplifying yields the following. [Ref. 6]

$$\left(\Delta \mathrm{H}_F\right)_\phi = \frac{M_\phi \rho_1 \Delta z}{j\omega\mu_\circ 4\pi}\Big\{2\left(V_3 + jkV_2\right) + \left(\rho_1^2 + \rho_2^2 + \left(z_2 - z_1\right)^2\right)\left(k^2V_3 - 3jkV_4 + 3V^5\right)$$

$$+ \rho_1\rho_2\left(3W_5 + 3jkW_4 - k^2W_3 + 3U_5 + 3jkU_4 - k^2U\right)\Big\} \qquad (72)$$

Two additional methods (Method B--Maxwell's curl equations applied twice to original vector potentials and Method C--solving $\overline{\Delta E_A}$ and $\overline{\Delta \mathrm{H}}$ directly from vector potentials by an alternative method) for determining the cross-field terms are derived in Appendix-A.

III. CALCULATING FORWARD PROPAGATION FIELDS AND TESTING THEIR CONVERGENCE

In this chapter, the ΔE and ΔH segments are used to generate a transfer matrix. The fields on a cylindrical surface at a distance of ρ_2 from the source can be determined by applying this transfer matrix to the equivalent currents generated by fields at a distance of ρ_1 from the source. The convergence of the three methods for determining ΔE and ΔH is examined. Choosing Method A, a parametric analysis of the convergence of fields at ρ_2 is conducted. Trends of the convergence of the calculated fields to the exact fields at ρ_2 are examined based upon differing input parameters in Method A.

A. CALCULATION OF TRANSFER MATRIX FOR CALCULATING THE ρ_2 FIELDS

Each $\overline{E_A}$, $\overline{E_F}$, $\overline{H_A}$, and $\overline{H_F}$ segment depends on several parameters, the most important of which is the distance R between the observation point on a cylindrical surface at a distance ρ_2 from the source and points on a cylindrical surface at ρ_1. In order to estimate the fields at the i^{th} point on ρ_2, the contributions from all points on ρ_1 must be summed. Figure 10 shows the physical relationship between the points on ρ_2 and the points on ρ_1.

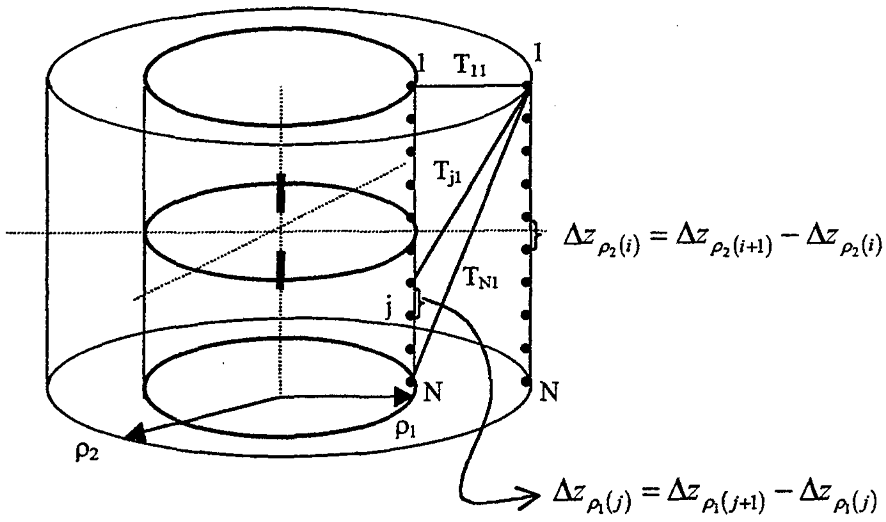


Figure 10. The Physical Relationship Between Points on ρ_2 and ρ_1

This implies a vector multiplication structure (as follows).

$$\begin{bmatrix} E_1 & E_2 & \dots & E_{N-1} & E_N \end{bmatrix}_{\rho_1} \begin{bmatrix} T_{1i} \\ T_{2i} \\ \vdots \\ T_{Ni} \end{bmatrix}_{\rho_1 \to \rho_2} = \begin{bmatrix} E_i \end{bmatrix}_{\rho_2} \tag{73}$$

For multiple points on ρ_2, the same vector of fields at points on ρ_1 is used but each point on ρ_2 will have an individual transform vector as shown below.

$$\begin{bmatrix} E_1 & \dots & E_N \end{bmatrix}_{\rho_1} \begin{bmatrix} T_{1(i-1)} \\ \vdots \\ T_{N(i-1)} \end{bmatrix}_{\rho_1 \to \rho_2} = \begin{bmatrix} E_{(i-1)} \end{bmatrix}_{\rho_2} \tag{74}$$

$$\begin{bmatrix} E_1 & \dots & E_N \end{bmatrix}_{\rho_1} \begin{bmatrix} T_{1i} \\ \vdots \\ T_{Ni} \end{bmatrix}_{\rho_1 \to \rho_2} = \begin{bmatrix} E_i \end{bmatrix}_{\rho_2} \tag{75}$$

$$[E_1 \;\; \cdots \;\; E_N]_{\rho_1} \begin{bmatrix} T_{1(i+1)} \\ \vdots \\ T_{N(i+1)} \end{bmatrix}_{\rho_1 \to \rho_2} = [E_{(i+1)}]_{\rho_2} \tag{76}$$

These vector multiplication equations can be condensed into a vector-matrix multiplication as shown below.

$$[E_1 \;\; \cdots \;\; E_N]_{\rho_1} \begin{bmatrix} T_{11} & \cdots & T_{1i} & \cdots & T_{1N} \\ \vdots & & \vdots & & \vdots \\ T_{N1} & \cdots & T_{Ni} & \cdots & T_{NN} \end{bmatrix}_{\rho_1 \to \rho_2} = [E_1 \;\; \cdots \;\; E_N]_{\rho_2} \tag{77}$$

$$E_{\rho_1}[T]_{\rho_1 \to \rho_2} = E_{\rho_2} \tag{78}$$

If the point distributions on ρ₁ and ρ₂ are uniform, then all $\Delta z_{\rho_1(j)} = \Delta z_{\rho_1}$ and all $\Delta z_{\rho_2(i)} = \Delta z_{\rho_2}$. If the point distributions are uniform and equivalent for both ρ_1 and ρ_2, then $\Delta z_{\rho_1} = \Delta z_{\rho_2} = \Delta z$. If either of the distributions on the ρ_1 or ρ_2 surfaces is not uniformly spaced, then the resultant transfer matrix **T** will have no commonly identifiable structure. If the point distributions are both uniformly spaced, but with $\Delta z_{\rho_2} \neq \Delta z_{\rho_1}$, then **T** will have some structure. When $\Delta z_{\rho_2} = \Delta z_{\rho_1}$, then the transfer matrix will be banded and have a Toeplitz structure [Figure 11][Ref. 7], with $t_n \neq t_{-n}$. If $\Delta z_{\rho_2} = \Delta z_{\rho_1}$ and $z_{\rho_2} = z_{\rho_1}$, then the transfer matrix will have a Toeplitz structure with $t_n = t_{-n}$.

$$T_{toeplitz} = \begin{bmatrix} t_0 & t_1 & \cdots & \cdots & t_{N-1} & t_N \\ t_{-1} & t_0 & t_1 & & & t_{N-1} \\ \vdots & t_{-1} & t_0 & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ t_{-(M-1)} & & & \ddots & \ddots & t_1 \\ t_{-M} & t_{-(M-1)} & \cdots & \cdots & t_{-1} & t_0 \end{bmatrix}_{(M+1)x(N+1)}$$

Figure 11. General Structure of a Toeplitz Marix

The bands in the transfer matrix are caused by the $\left(z_{\rho_2 m}-z_{\rho_1 n}\right)^2$ terms in R, $(\Delta E)_z$ and ΔH, and $\left(z_{\rho_2 m}-z_{\rho_1 n}\right)^1$ terms in $(\Delta E)_\rho$. When Δz's are uniform and equal, but the actual z distributions are not equivalent (i.e. offset from each other), then the transfer matrix elements are translation invariant and depend only on m-n. When Δz's are uniform and the z distributions are aligned, then only the $(\Delta E)_\rho$ term is translationally invariant depending on m-n. All the other terms are based on $\left(z_{\rho_2 m}-z_{\rho_1 n}\right)^2$, so they are translationally invariant depending on |m-n|. Transfer matrices with the m-n dependence are full, nonsymmetric Toeplitz matrices, $t_n \neq t_{-n}$ in Figure 11. Transfer matrices with the |m-n| dependence are full, symmetric, non-Hermetian Toeplitz matrices, $t_n = t_{-n}$ in Figure 11. An interesting side note is that transfer matrices for $\Delta z_1 = C\Delta z_2$ or $\Delta z_1 D = \Delta z_2$, with C and D integer constants greater than one, are made by selecting specific rows or columns of a Toeplitz matrix based on the Δz with the smallest size.

The specific fields required for this thesis' study of back propagation are the E_z and H_ϕ fields on the ρ_2 cylindrical surface. These total fields are found using a large (2Mx2N) transfer matrix consisting of four blocks. Each block is the (MxN) transfer matrix for a specific contribution to the total field from one of the two equivalent currents on the ρ_1 cylindrical surface.

$$T_{2Mx2N} = \begin{bmatrix} T_{EE} & T_{EH} \\ T_{HE} & T_{HH} \end{bmatrix} \qquad (79)$$

$$\lfloor E_z H_\phi \rfloor_{\rho_1} T = \lfloor E_z H_\phi \rfloor_{\rho_2} \qquad (80)$$

In general the complex elements of T_{HE} are very large in magnitude compared to the rest of the matrix elements. Conversely, the elements of T_{EH} are very small compared to the other elements. The relative magnitude of the T_{EE} and T_{HH} elements are roughly the same. This difference in relative magnitudes is due the small H-field at ρ_1 contributing to the large E-field at ρ_2 for T_{HE}, and, conversely, the large E-field at ρ_1 contributing to the small H-field at ρ_2 for T_{EH}.

Additionally, T will generally be rectangular since more points will be determined at ρ_2 than are measured at ρ_1. This is in preparation for over determined inverse systems.

B. COMPARISON OF THREE METHODS FOR DETERMINING THE CALCULATED FIELDS AT ρ_2

The three different methods for determining the calculated fields at ρ_2 are compared using the Least Squares Error (LSE) of each method's field compared to the exact field.

$$LSE_i = \sqrt{\frac{\sum\left(F_{EXACT} - \hat{F}_{CALC,i}\right)^2}{\sum\left(F_{EXACT}\right)^2}} x100 \qquad (81)$$

The LSE is a measure of a particular method's total energy contained in the error with respect to the total energy contained in the exact field, expressed as a percentage. A comparison of the three methods LSE shown in Figure 12 reveals that their differences were indistinguishable. This indicates that no particular method is better than the others in the near field. Method A was chosen for all further investigations. For the fields calculated in Figure 12 and in all subsequent figures, the frequency of the current distribution on the dipole is $f = 300$ MHz. This frequency corresponds to a wavelength $\lambda = 1$ meter. Thus, values of ρ and z also represent values relative to λ.

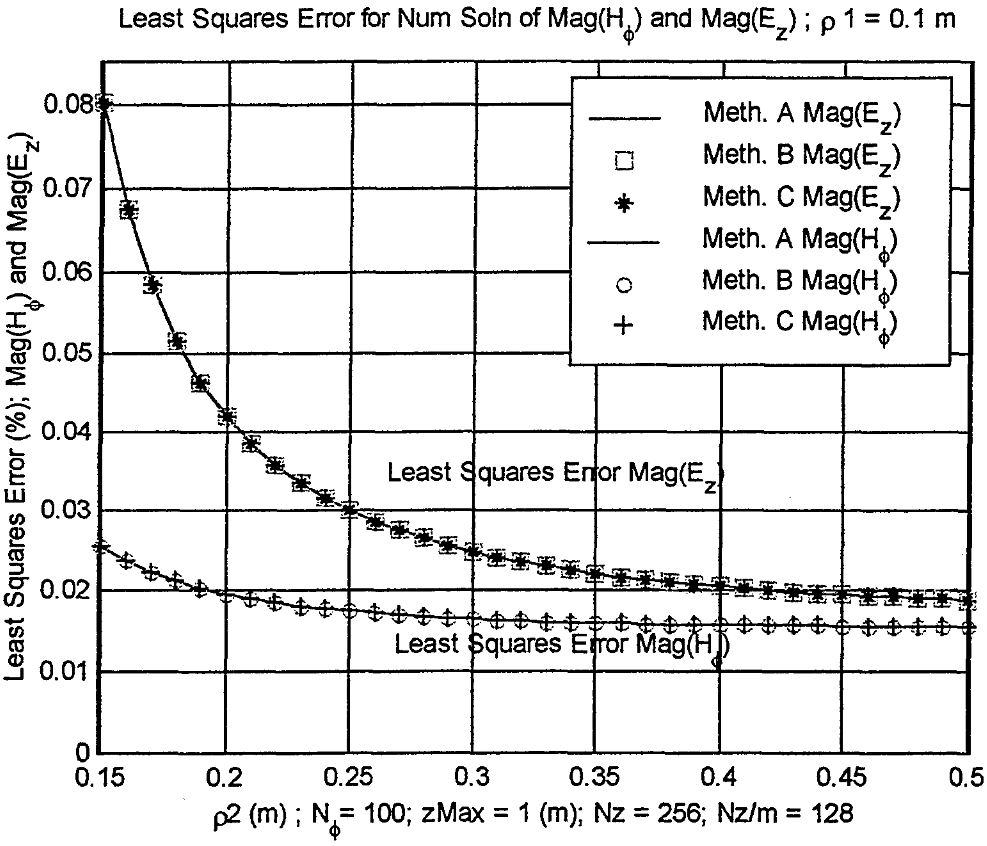


Figure 12. LSEs for Three Methods on the Same Plot

C. CUMULATIVE TOTAL ENERGY DENSITY OF FIELDS

In order to get an idea about the spread of energy contained in the fields of a single dipole, the cumulative total energy distribution for both E_z and H_ϕ is determined in Figure 13 and Figure 14. For a dipole centered at the origin, the plots illustrate the percent of total field energy in the field for $|z| > z_c$, the z truncation. These plots provide a guideline for choosing a maximum z in order to keep a standard amount of field energy when doing calculations. Additionally, it is also a measure of the rapidness of change in percent of energy as z_c approaches zero. These two considerations provide a starting point for understanding the effects of changing parameters to the amount of error to be expected in the calculated fields at ρ_2.

D. PARAMETRIC ANALYSIS OF THE CALCULATED FIELDS USING METHOD A

The following subsections outline trends in the accuracy of the calculated fields with respect to the exact fields on the cylindrical surface at ρ_2. All calculations use uniformly distributed, equivalent point distributions for z_{ρ_1} and $z_{\rho 2}$, with the z_{ρ_1} maximum/minimum equal to the z_{ρ_2} maximum/minimum. The relative merit of various parameter changes can be reviewed using three techniques. First, the Real and Imaginary components of the exact and calculated fields are plotted together. Second, the Relative Error of the magnitude of the calculated field with respect to the magnitude of the exact field on a point to point basis is plotted. Last, the Least Squares Error expressed as a percentage is plotted.

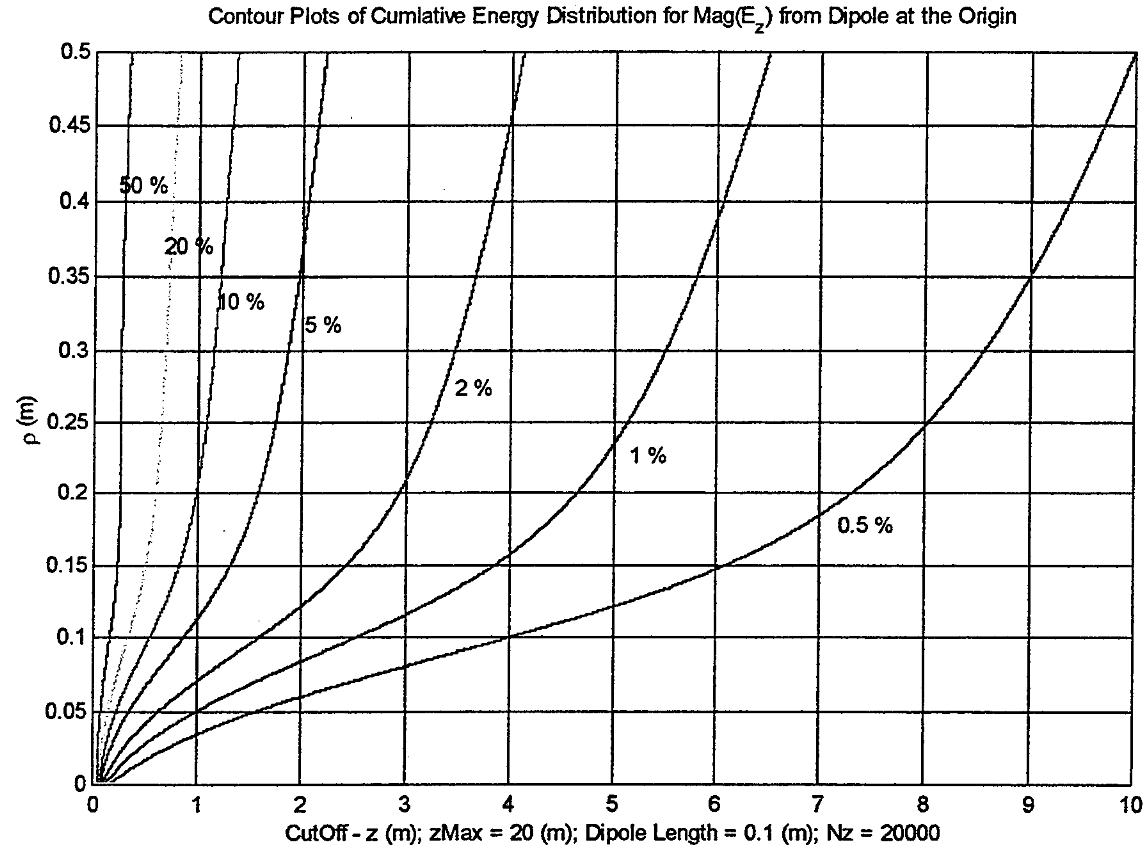


Figure 13. E Plot

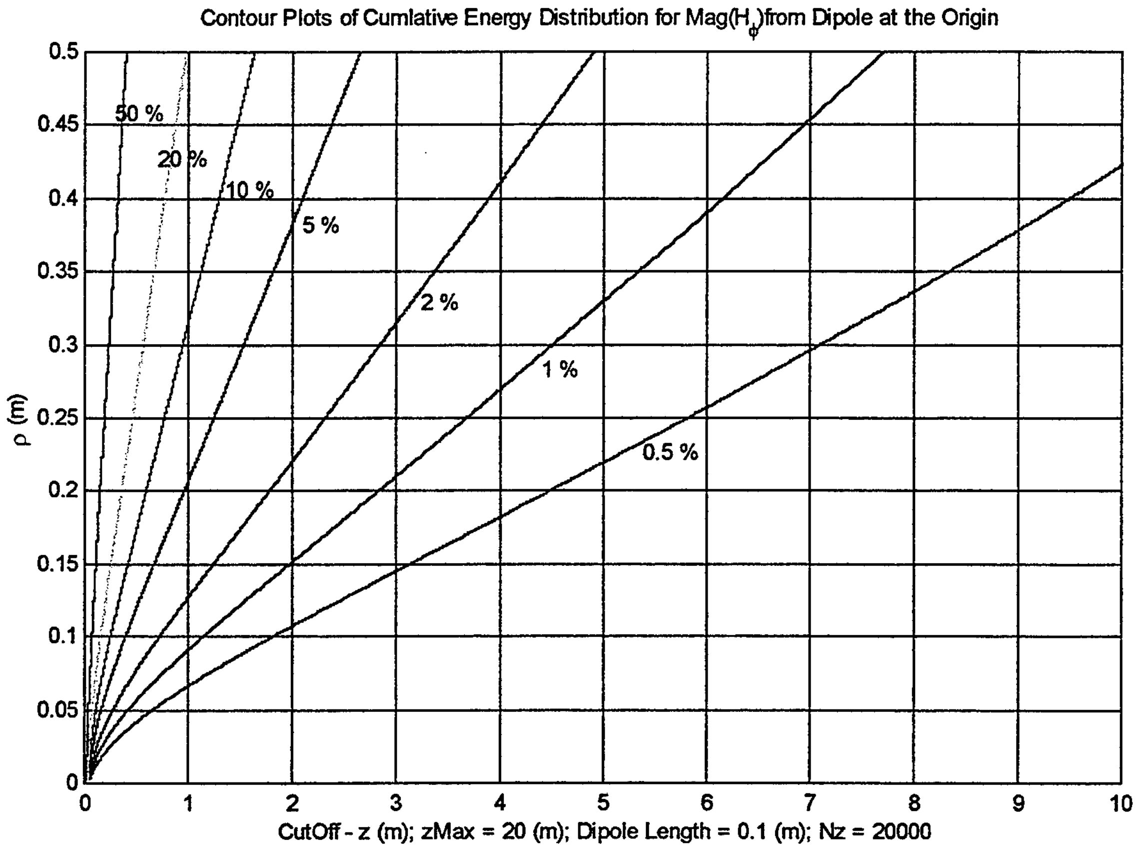


Figure 14. H Plot

1. Varying the Truncation in Z

To get an appreciation of the effect of varying the truncation of z at a fixed ρ value, Figure 15 shows a plot of the LSE in the calculated ρ_2 fields for a range of maximum z. From Figures 13 and 14 it is clear that as the cylindrical surface ρ_1 moves further from the source the value of the z cutoff point must be increased to maintain a constant level of the field's energy for ρ_2 calculations. Figure 13 shows that changes in the z truncation point have a significant impact on the amount of error in the ρ_2 fields when the magnitude of the maximum z is small. Then, beyond a certain point, increasing the truncation point in z does little to decrease the error in magnitude z ρ_2 fields. Figure 15 confirms these two observations. For z cutoff less then two meters, the LSE is greater than one percent.

2. Varying the Length of Δz

In building the transfer matrix and defining the fields on the two cylindrical surfaces, an estimation of the actual smooth continuous fields was made. The fields were estimated to be the same over a length of z, called Δz. As Δz decreases, higher spatial frequencies for fields can be resolved uniquely. In other words, the estimate of the continuous field made by discrete steps Δz apart becomes more accurate as Δz decreases. If there are no spatial frequencies higher than some k_z, then decreasing Δz when $\Delta z << \frac{2\pi}{k_z c}$ will provide little improvement to the calculated field accuracy. Conversely, decreasing Δz when $\Delta z >> \frac{2\pi}{k_z c}$ will dramatically improve the field accuracy.

These trends are easily seen for cases when the z truncation point is large enough to include most of the energy in the field. As seen in Figure 16, when coarse estimates of the field are refined, large Δz is decreased and the error in the calculated fields also decreases.

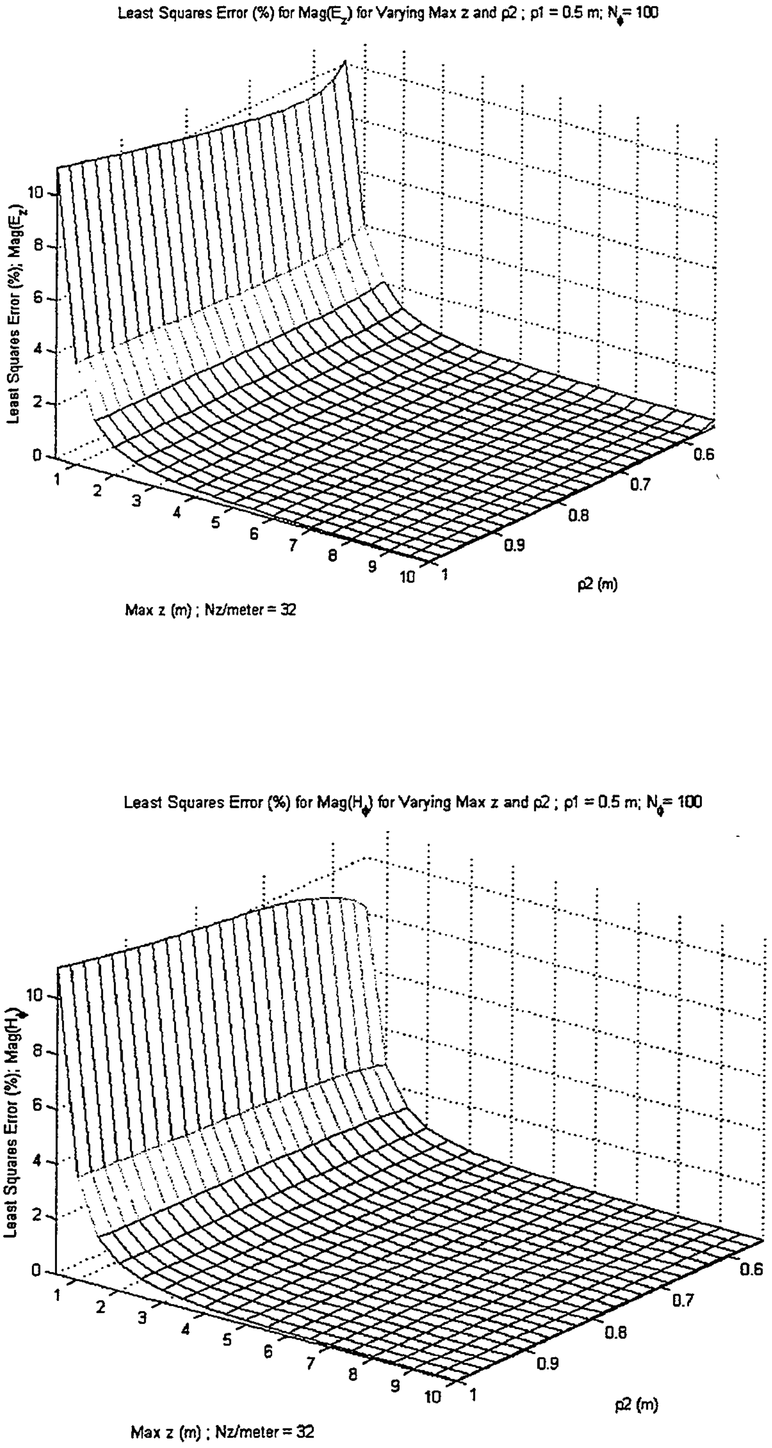
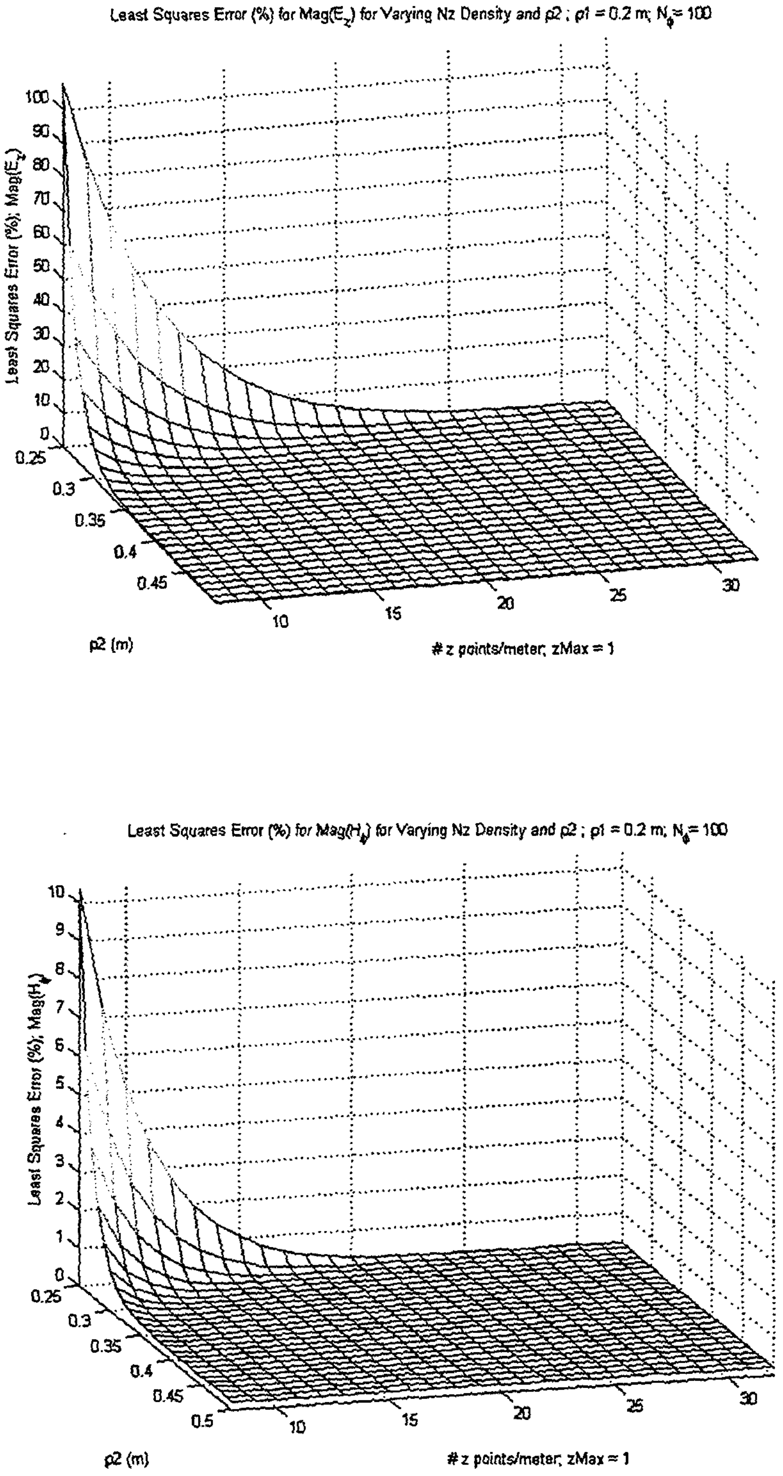


Figure 15. Plots of Varying Maximum z



Least Squares Error (%) for Mag(E_z) for Varying Nz Density and $\rho 2$; $\rho 1 = 0.2$ m; $N_\phi = 100$

Least Squares Error (%) for Mag(H_ϕ) for Varying Nz Density and $\rho 2$; $\rho 1 = 0.2$ m; $N_\phi = 100$

Figure 16. Plots of E_z and H_ϕ for Varying Δz

Two special cases exist, both of which are caused by rapid fluctuations and large estimates obtained when R becomes small, and $\frac{1}{R^n}$ becomes large. The first case occurs throughout the region of interest when $\Delta\rho=\rho_2-\rho_1$ becomes small. This causes the calculated fields to be very large and locally driven, even when these calculated fields are well behaved for a ρ_2 only a few more $\Delta\rho$ lengths away. Decreasing the size of Δz is required to compensate for this effect as seen in Figure 17. The second case occurs when both ρ_1 and ρ_2 are small in general. This means that fields close to the source are being estimated. These fields have rapid variation in the z direction and have large magnitudes as well. Additionally, since ρ_1 is small and ρ_2 is small, $\rho_1-\rho_2$ will also be small, thus exacerbating the error by including large calculated fields. Figure 18 shows that decreasing the size of Δz again helps to counteract this effect.

3. Varying the Number of Thin Ring N_ϕ Segments

Varying the number of segments (N_ϕ) on the ring leads to changes in the segment size. The segment size is equal to $\rho_1\Delta\phi_1$.

$$\rho_1\Delta\phi_1=\frac{2\pi\rho_1}{N_\phi} \qquad (82)$$

As N_ϕ increases, the length of the segment size decreases and a more accurate calculation for U_n, V_n, W_n, and the fields is produced. As an initial estimate, the segment size should be small compared to λ_o of the testing frequency. A more conservative limit on segment size would be to make it less than or equal to the size of Δz. Alternatively, the segment size can be limited to between a fifth and a tenth of the size of $\Delta\rho=\rho_2-\rho_1$. This limit provides insight into why segments should be at least this small if not smaller. As ρ_2 gets closer to ρ_1, the points most influenced by the thin ring and by the ring segments become close to the center of the actual segment as shown in Figure 19.

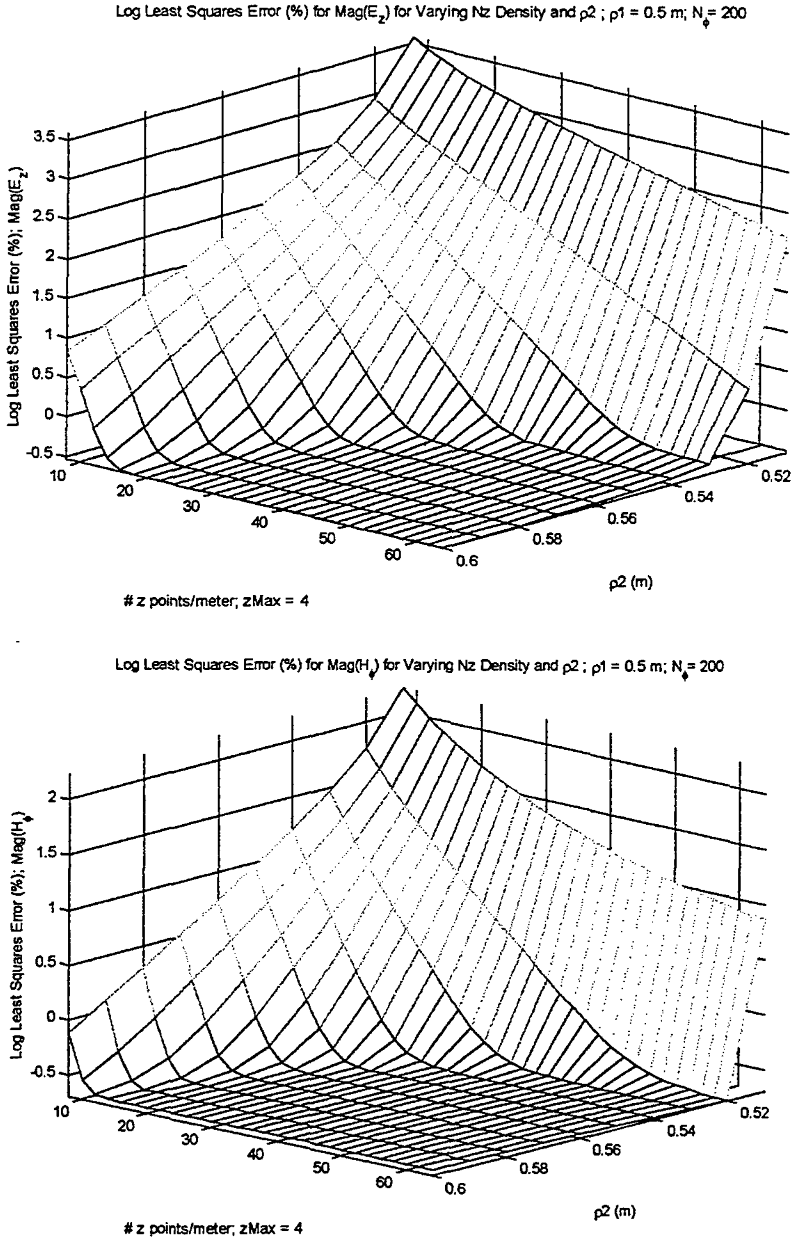
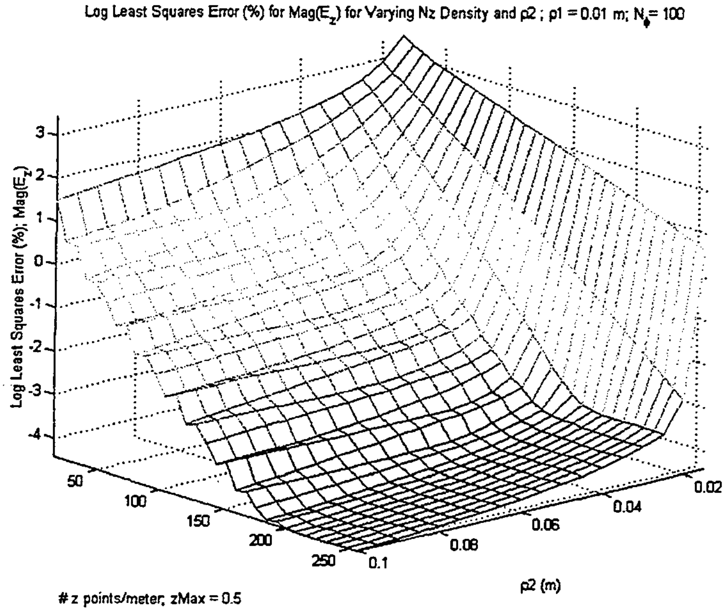


Figure 17. Case I: Mid range ρ_1, with Small $\Delta\rho$ and Moderate $\Delta\rho$



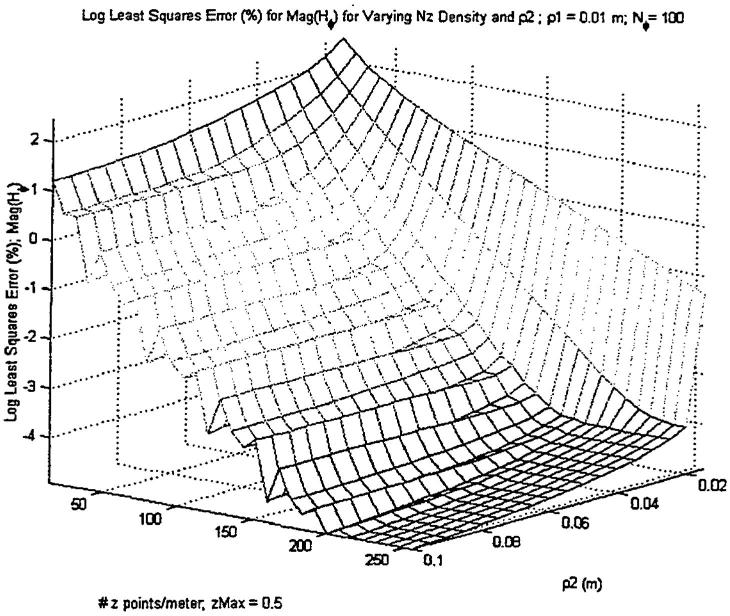
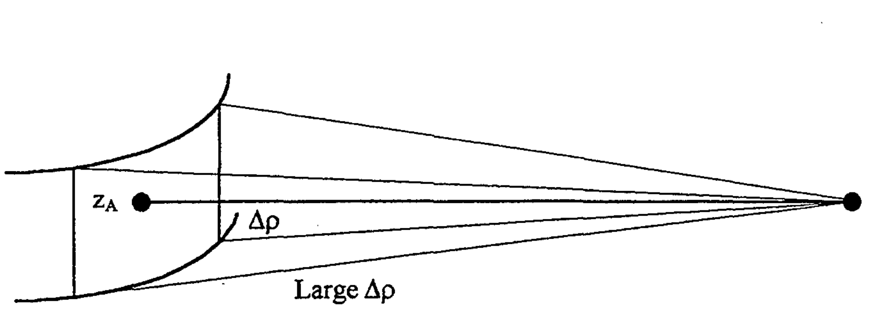
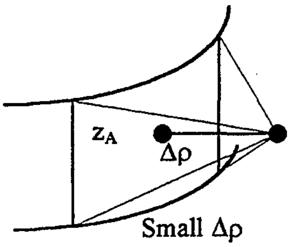


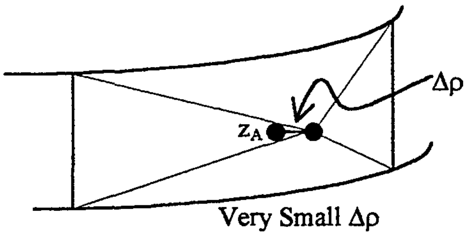
Figure 18. Case II: ρ_1 Close to Zero, with Small $\Delta\rho$ and Moderate $\Delta\rho$



Segment Area



Large Segment Area



Large Segment Area

Figure 19. Effect of Segment Size on Error

If the overall size is not reduced, then that segment's calculated value is not a good estimate for the field contributions it represents. Figure 20 illustrates the effects of small N_ϕ versus large N_ϕ, and large segment size versus small segment size, respectively. Clearly, for small N_ϕ, the calculated fields are a poor estimate. Increasing N_ϕ quickly improves estimates. For further calculations the segment length will be ~Δz for small

Δz. This gives $N_\phi = \frac{2\pi\rho_1}{\Delta z}$ for small Δz. For all calculations used in the plots, N_ϕ is set to $\frac{2\pi\rho_1}{\Delta z}$ or 100, whichever is the larger of the two.

4. Overall Trends

From a review of the previous figures, several other interesting features can be discussed. First, the calculated fields all have errors close to the z truncation points. This is clearly shown on the plots of Relative Error. This is caused by not including the influence on ρ_1 at points $|z| >$ z cutoff. By truncating fields greater than a certain z maximum, error is introduced. The error at the edges of the calculated field is due to the proximity of field points on ρ_1 that are not included when making an estimation of ρ_2 fields.

Another less obvious effect occurs when increasing ρ_2 from a fixed ρ_1. After an initial dip in error, LSE plots show error slowly increasing again as ρ_2 increases.

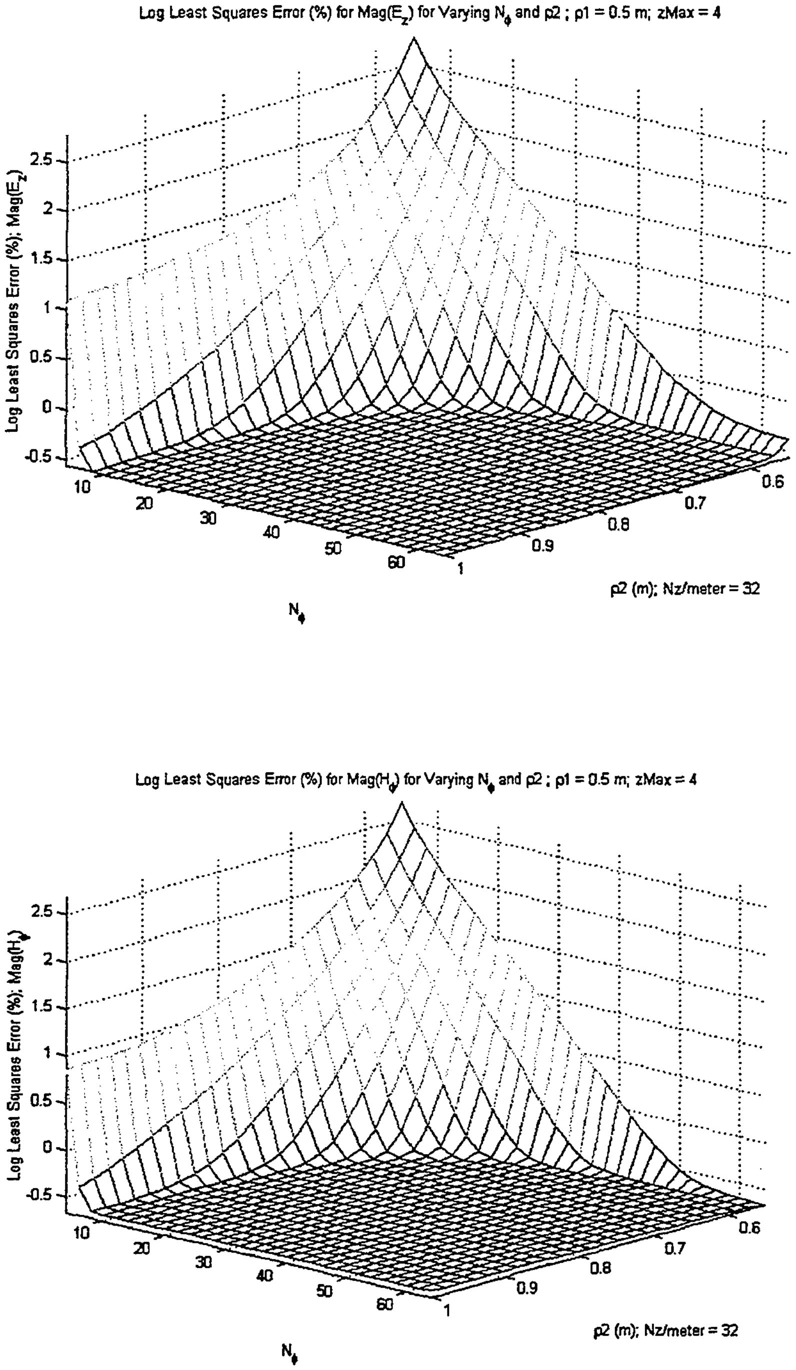


Figure 20. Plots Showing Effect of Small Versus Large N_ϕ

THIS PAGE INTENTIONALLY LEFT BLANK

IV. BACK PROPAGATION OF FIELDS

In forward propagation of fields, the contributions of thin rings of electric and magnetic equivalent currents determined from the fields at the ρ_1 cylindrical surface were summed to produce an estimate of the exact fields at the ρ_2 cylindrical surface. This process was described mathematically in the form of vector-matrix multiplication.

$$\left[E_z H_\phi\right]_{\rho_1} \begin{bmatrix} T_{EE} & T_{EH} \\ T_{HE} & T_{HH} \end{bmatrix} = \left[E_z H_\phi\right]_{\rho_2 CALCULATED} \qquad (83)$$

Algebraically, to obtain the fields E_z and H_ϕ at ρ_1 from knowledge of the calculated fields at ρ_2, it appears that multiplying both sides of the equation by the inverse of the transfer matrix T will yield the correct answer.

$$\left[E_z H_\phi\right]_{\rho_1} [T]\,[T]^{-1} = \left[E_z H_\phi\right]_{\rho_2} [T]^{-1} \qquad (84)$$

$$\left[E_z H_\phi\right]_{\rho_1} = \left[E_z H_\phi\right]_{\rho_2} [T]^{-1} \qquad (85)$$

Figure 21 shows the effect of T^{-1} on the calculated E_z and H_ϕ fields of ρ_2. The reformed E_z and H_ϕ fields of ρ_1 tend to be very close to the exact ones. In fact, the difference is within machine accuracy in some areas. This is expected since the T matrix actually formed the calculated fields.

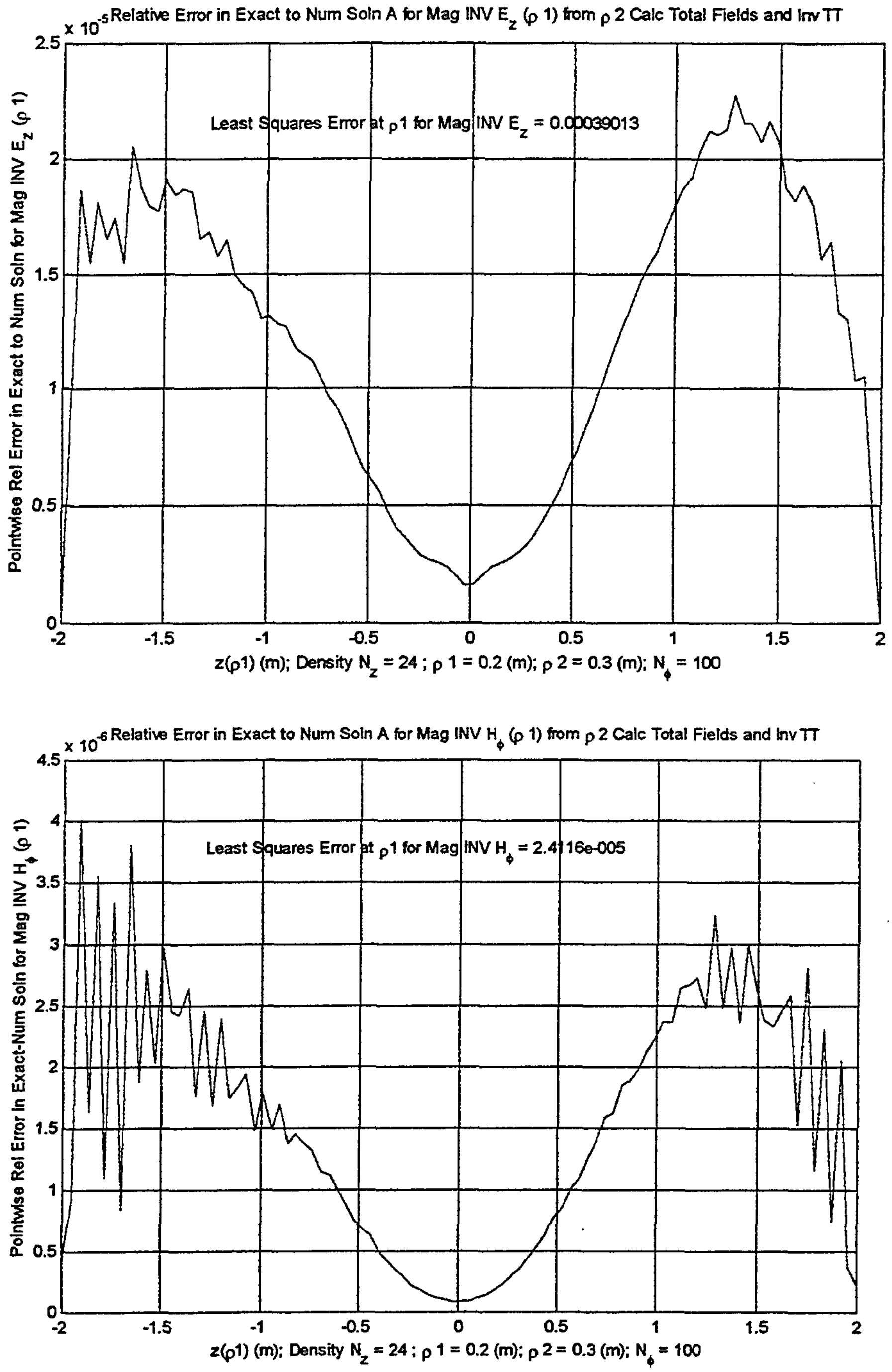


Figure 21. Effect of T^{-1} on the Calculated E_z and H_ϕ Fields of ρ_2.

A. EFFECT OF APPLYING INVERSE T MATRIX TO THE EXACT FIELDS AT ρ_2

When the calculated $\left[E_{z_2} H_\phi\right]_{\rho_2}$ is replaced by the exact $\left[E_{z_2} H_\phi\right]_{\rho_2}$, a well behaved T^{-1} would yield newly calculated inverse fields close to the source fields at ρ_1. If the difference between the calculated and exact fields at ρ_2 is small, then applying a well behaved T^{-1} would yield a newly calculated field whose values would be close to the exact ρ_1 field values. Mathematically, the following relationship holds true for a well behaved T^{-1} and δ_{INVT} is small.

$$\left[E_z H_\phi\right]_{\rho_1} + \delta_{INVT} = \left(\left[E_z H_\phi\right]_{\rho_{2CALC}} + \delta_{\Delta CALCtoEXACT}\right)\left[T\right]^{-1} \qquad (86)$$

Unfortunately, T^{-1} is not well behaved in this case. Inverse T is extremely sensitive to small changes in the fields at ρ_2 used as its input. As can be seen from Figure 22, even a small change of δ_Δ between the calculated and exact fields at ρ_2 will cause large differences between the exact and inverted fields at ρ_1. At first glance, it seems that the best way to get better results is to make the small change in δ_Δ even smaller. Figures 23 and 24 show the pointwise relative error of the magnitude of the inverted fields compared to the exact fields at ρ_1. Two cases are shown in each figure, one for a segment density of eight per meter and the other for a segment density of sixteen per meter. The LSE for the inverted ρ_1 fields and the calculated ρ_2 fields is also shown to give a measure for the overall improvement increasing segment density provides. Also the maximum δ_Δ between calcualted and exact fields at ρ_2 is provided.

Part of the reason for the poor improvement in field esstimation lies with T and T^{-1}. They are classically ill-conditioned matrices. Small changes at the input of an ill-conditioned matrix cause large changes at the output. One measure of this effect for a matrix is called the condition number. When condition numbers are very large, matrices are considered ill-conditioned. Additionally, the transfer matrix becomes physically larger when either more terms are used to determine ρ_2 fields or when more ρ_2 field points are desired. Therefore, as the accuracy of the calculated fields improves, more points have been used and T becomes larger and larger. If T is an ill-conditioned matrix, Figure 25 shows that larger versions of T are even more ill-conditioned. The growing condition number is one reason that decreasing δ_Δ has only a small improvement in the accuracy of the calculated inverse fields. Therefore, in order to get a better, smaller δ_Δ, the size of T must be increased, but in increasing T, T^{-1} is increased and becomes even more ill-conditioned, countering the effects of a smaller δ_Δ.

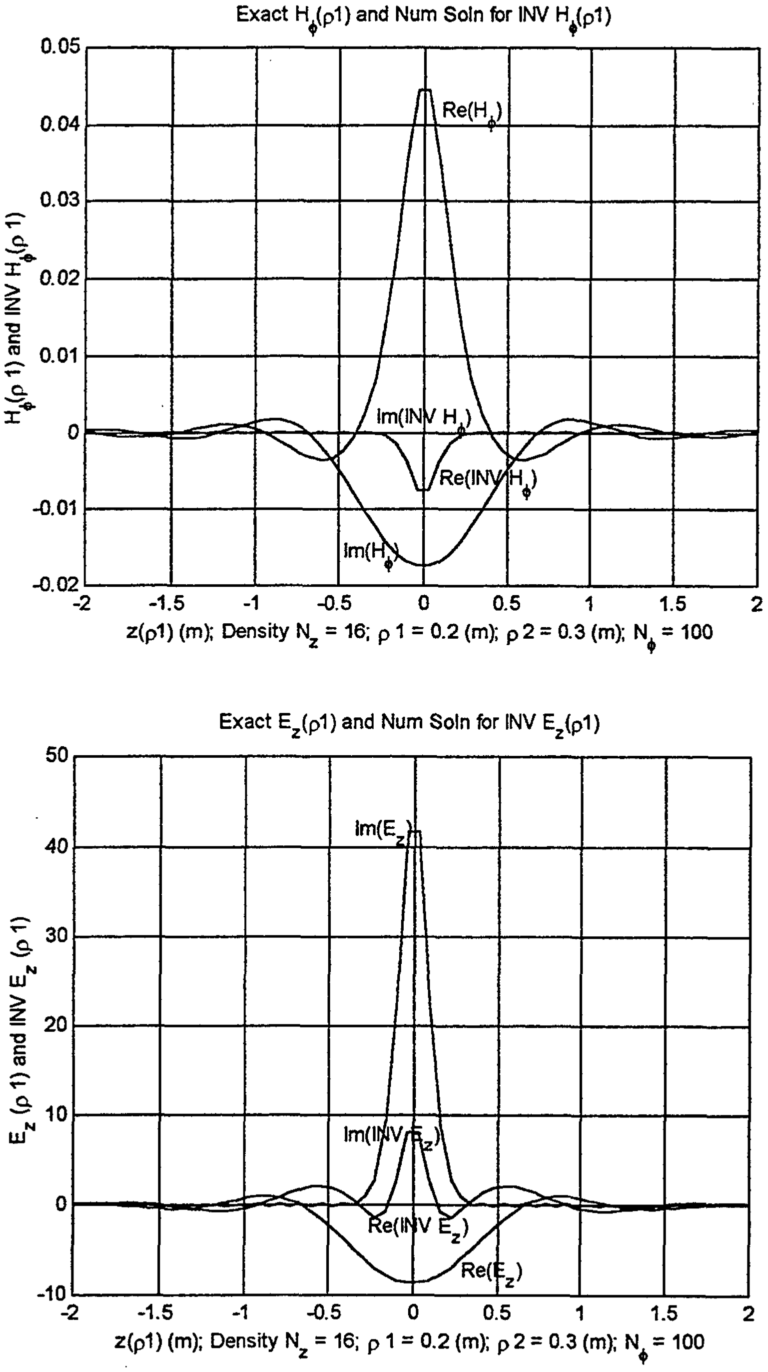


Figure 22. T^{-1} Sensitivity.

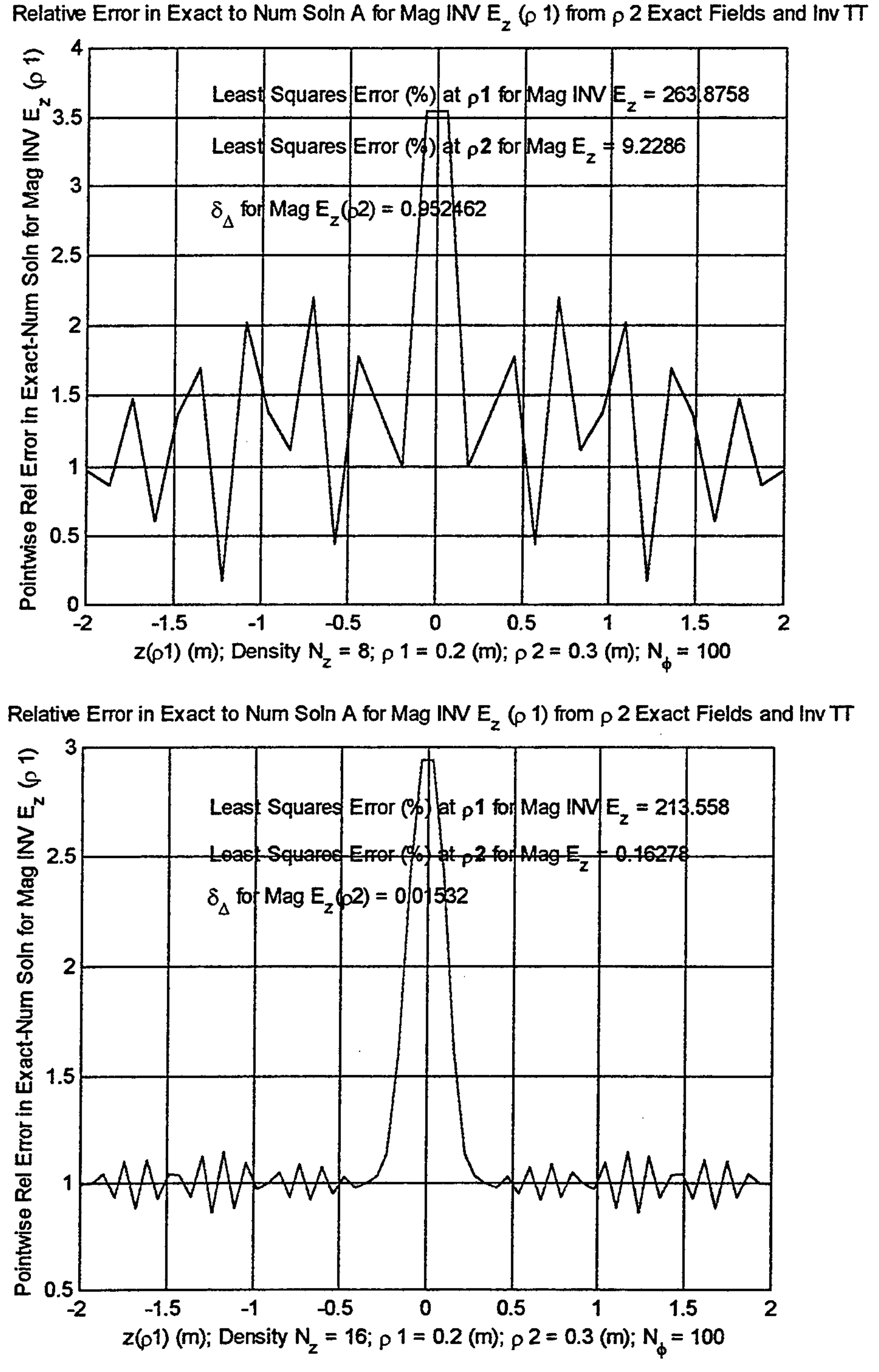


Figure 23. Effect of Decreasing δ_Δ on Error in Electric Fields

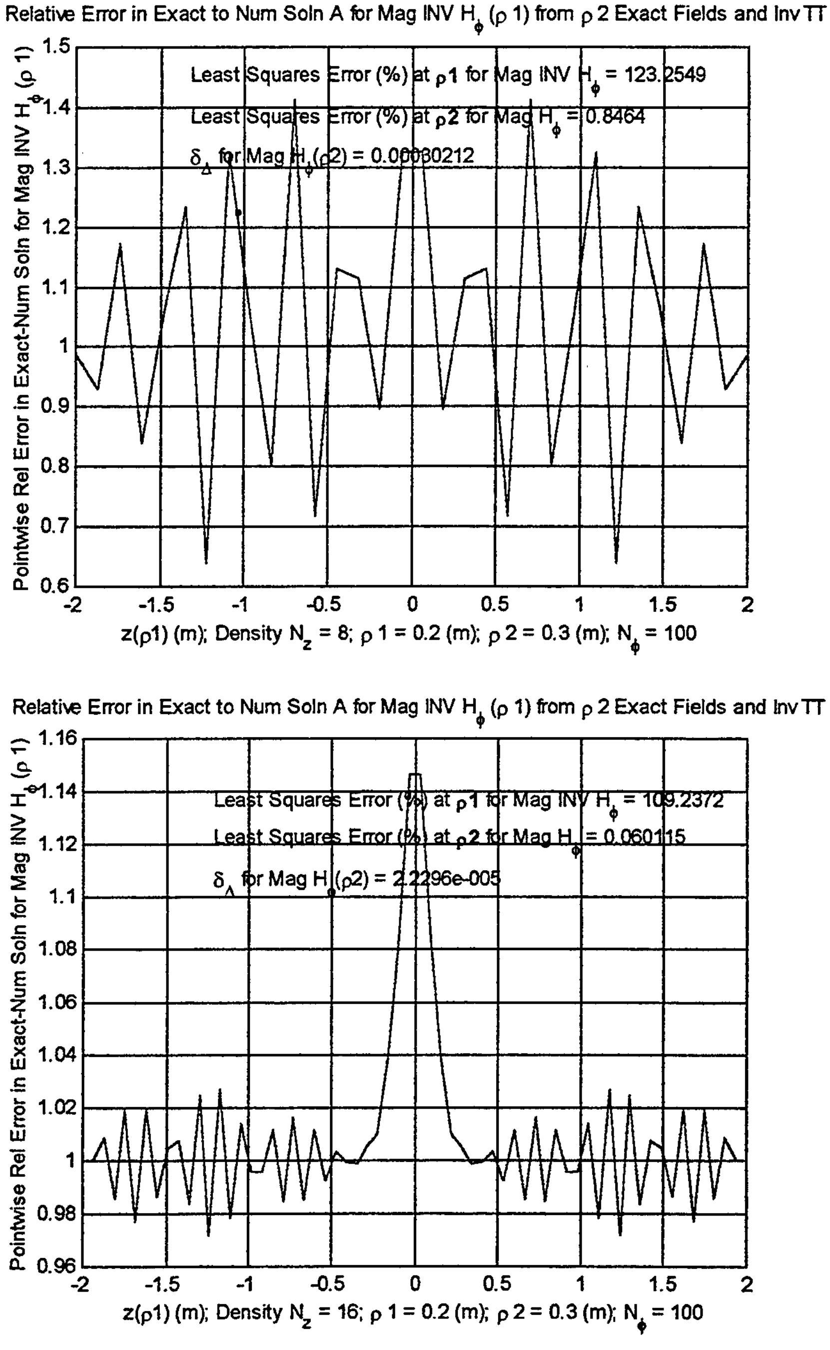


Figure 24. Effect of Decreasing δ_Δ on Error in Magnetic Fields

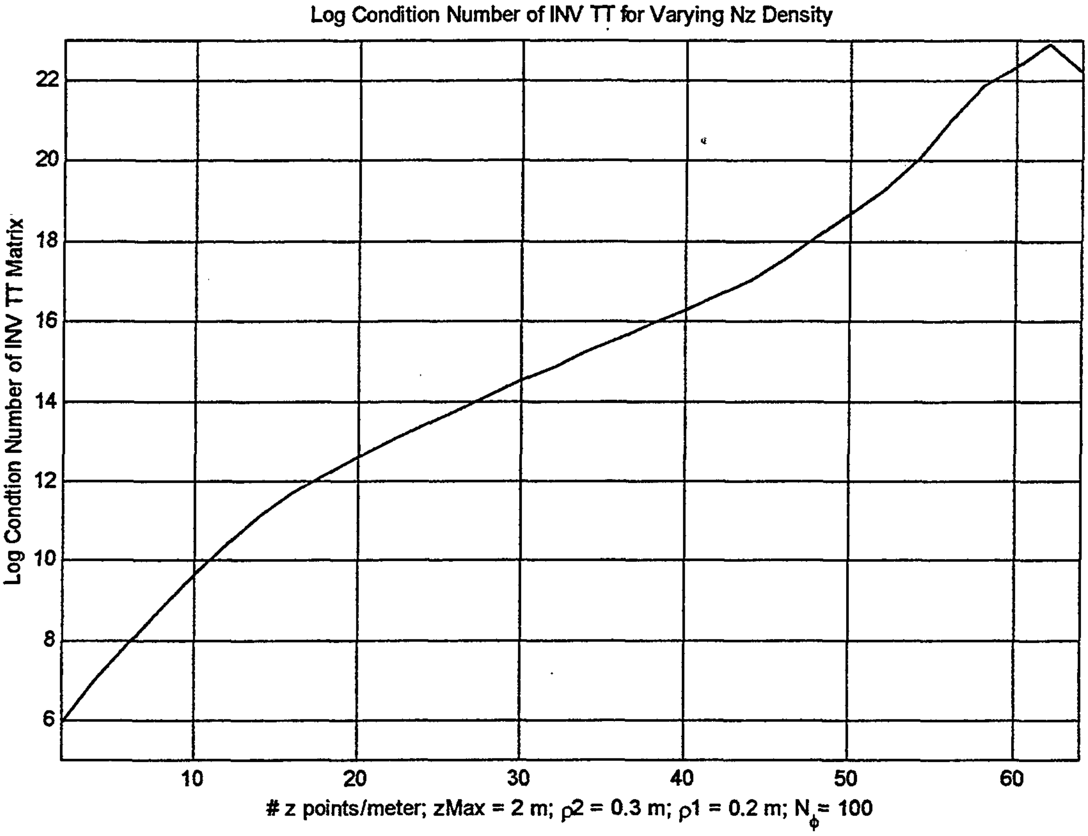


Figure 25. Condition Numbers of the Transfer Matrix

B. USING AN OVERDETERMINED INVERSE T MATRIX

In an attempt to improve the accuracy of the inverted fields, the inverse T matrix is constructed to be over determined. The fields at ρ_2 were determined at more points than the fields at ρ_1. This leads to a "long and skinny" T matrix as indicated by the following equation.

$$\lfloor E_z H_\phi \rfloor_{(1x2M)} [T]_{2Mx2N} = \lfloor E_z H_\phi \rfloor_{(1x2N)} \qquad (87)$$

$$N>M$$

Since T is no longer a square matrix, the inverse of T is not quite as straight forward. Instead of the standard A^{-1} of a nonsingular square matrix, the pseudoinverse, A^{+}, is calculated using the SVD technique shown in equation (88) where U and V are unitary matrices and S is a diagonal matrix. The pseudoinverse is calculated in equation 89 from the inverse of S, S^{+}.

$$A=USV' \qquad S=\begin{bmatrix} s_1 & & 0 \\ & s_2 & \\ 0 & & \ddots \end{bmatrix} \qquad (88)$$

$$A^{+}=VS^{+}U' \qquad S^{+}=\begin{bmatrix} \frac{1}{s_1} & & 0 \\ & \frac{1}{s_2} & \\ 0 & & \ddots \end{bmatrix} \qquad (89)$$

Note, when A is 2Mx2N, with N > M, then A^{+} is 2Nx2M and $AA^{+}=I_{(2Mx2M)}$. By using the SVD to find the pseudoinverse, the matrix T^{+} will find the least squares estimate of the ρ_1 Inverse Fields from the over determined inputs [Ref. 9]. Figure 26 shows the LSE inverted fields at ρ_1 when increasing the number of ρ_2 segments by multiples of the number of ρ_1 segments. After the ratio of ρ_2 to ρ_1 segment densities increases beyond three, very little improvement in the inverted fields' LSE is observed. In fact, the LSE degrades as the ratio increases from one to three.

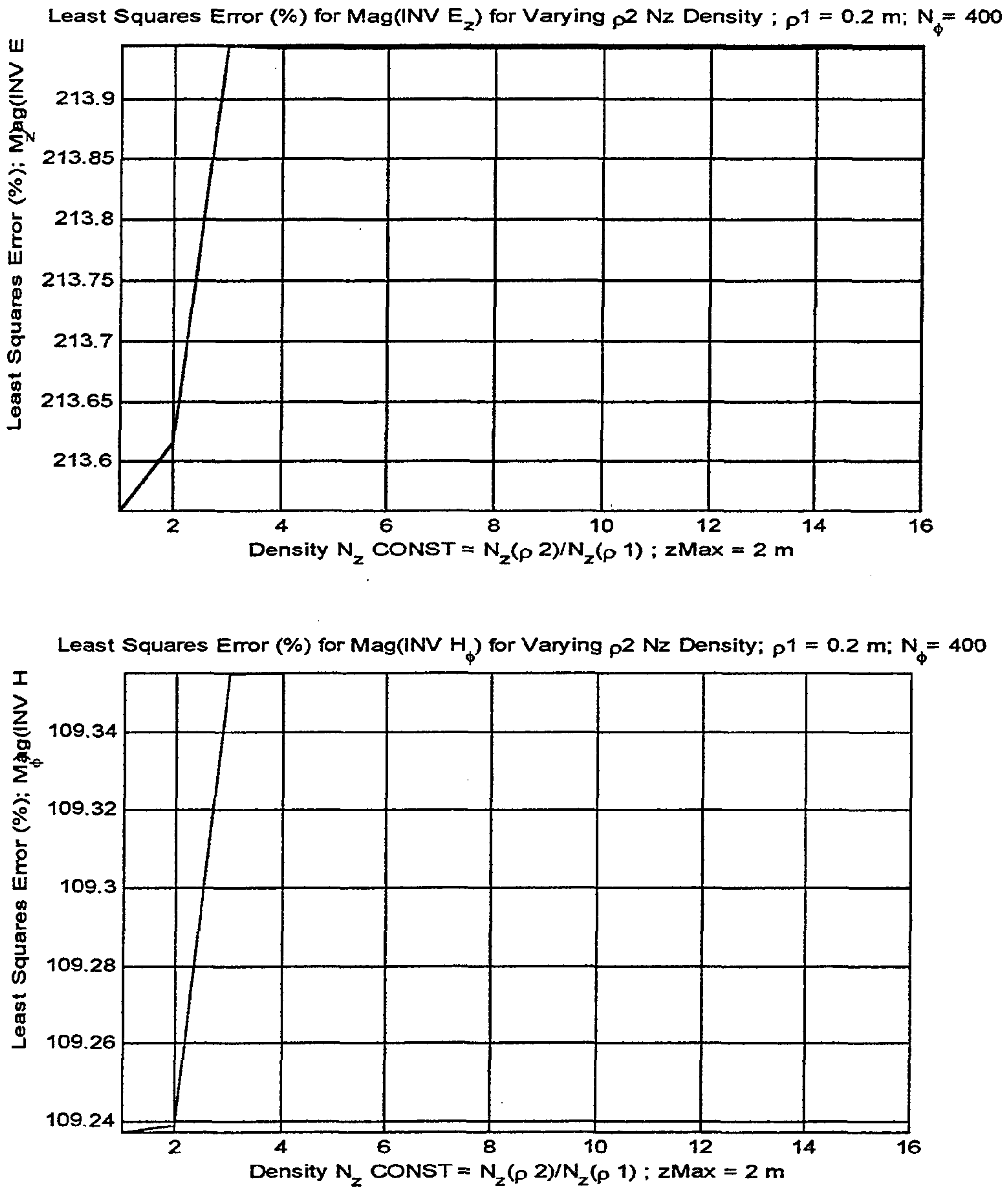


Figure 26. T^+ Does Not Improve the Calculated Inverse Fields at ρ_1.

C. IMPROVING INTEGRATION ESTIMATE ON ρ_1 WHILE MAINTAINING A SMALLER T MATRIX

The next attempt at improving the calculated inverse field is to maintain a smaller T matrix size while attempting to get some benefit from more integration accuracy. In order to do this, a new matrix D is defined such that it is a 2CN x 2M matrix created from a z distribution as shown in Figure 27. Then bands containing an odd number, C, of the matrix's rows are added together and divided by the number of rows in the band, C. This new averaged integration row is then used as a row in T, a 2N x 2M matrix, where it is multiplied by the field point corresponding to the original D matrix band's middle row.

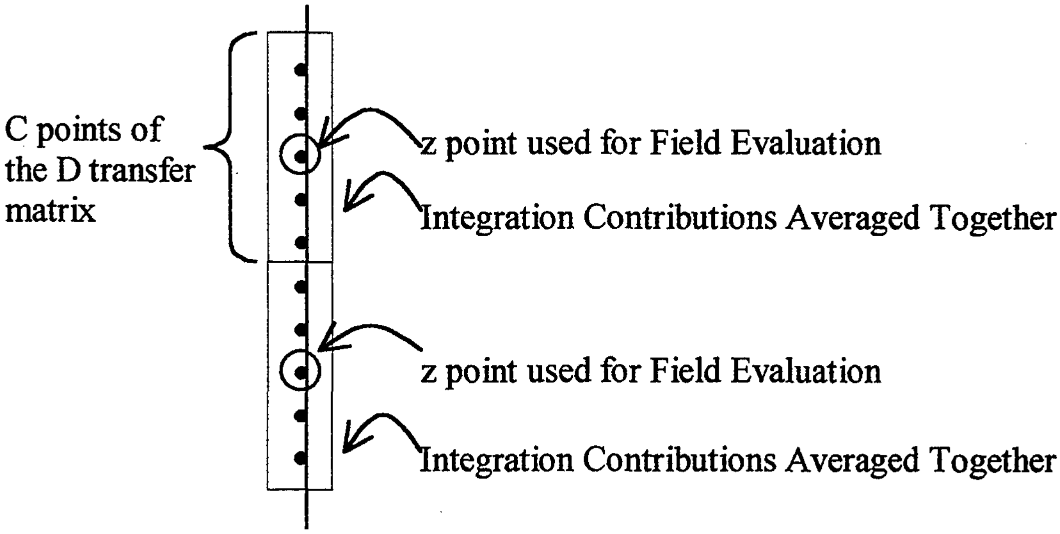


Figure 27. New Approximate Integration Row Used in T.

Thus, $[D]_{2CNx2M}$ becomes $[T]_{2Nx2M}$ by averaging bands of C rows, where C is and odd integer. Figure 28 shows the LSE of inverted fields at ρ_1 for an increasing ratio of ρ_1 segment density used to improve the integration accuracy to the ρ_2 segment density. As the ratio increases beyond three, the inverted field LSE changes only slightly.

D. A CLOSER LOOK AT REAL AND IMAGINARY FIELD COMPONENTS

Figures 29 and 30 show the Real and Imaginary E-field components for the exact ρ_1 E-fields and the ρ_1 E-fields calculated using the matrix T inversion process for several ρ_1, ρ_2 pairs. Figures 31 and 32 show the Real and Imaginary H-field components for ρ_1 H-fields. It is striking that in all cases, the Real Inverted E-fields and the Imaginary Inverted H-fields are essentially zero.

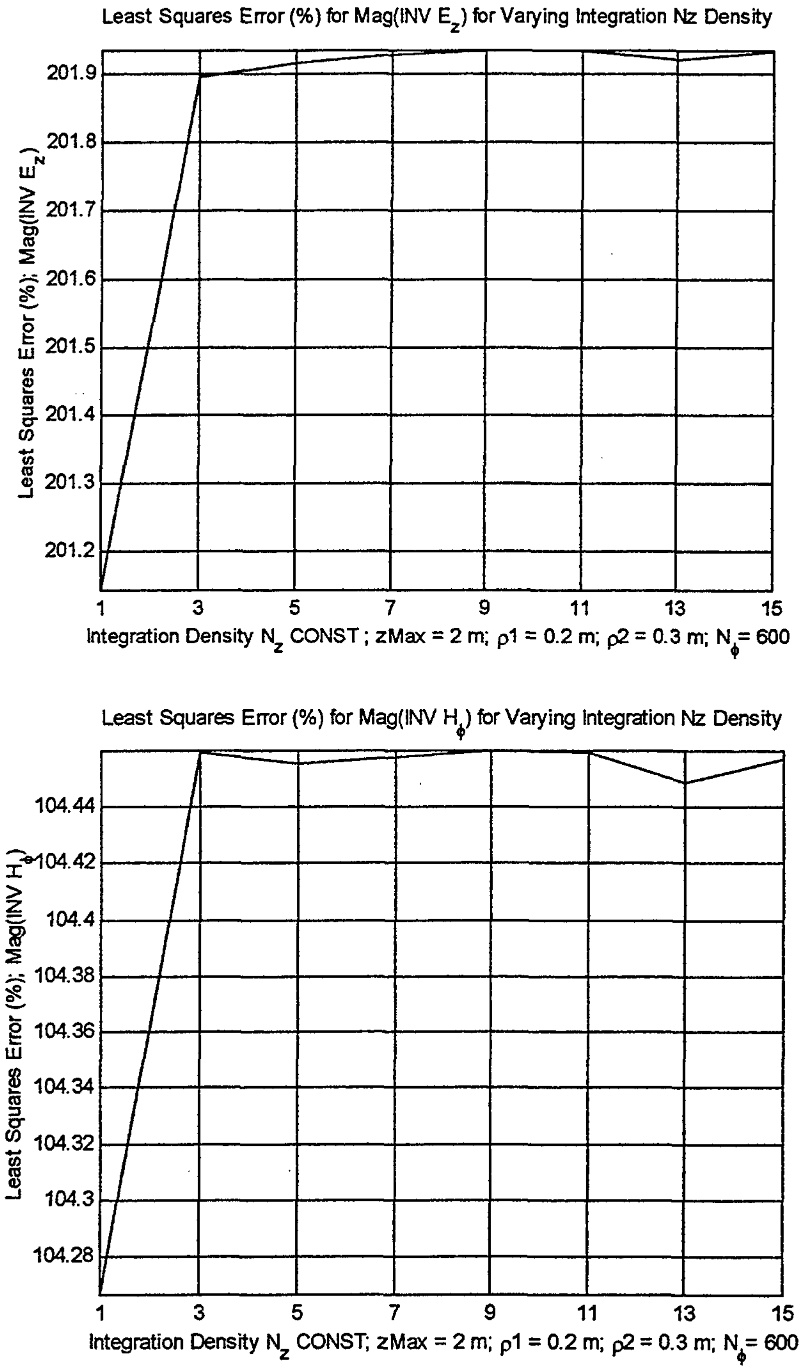


Figure 28. Flat Weighted Integration—Inverse Fields

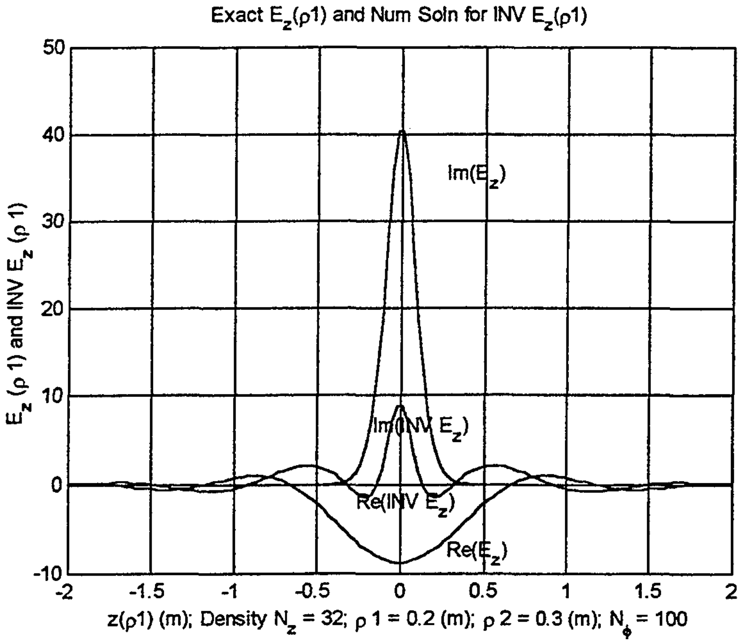


Figure 29. Back-Propagated and Exact Electric Field Components, Case 1

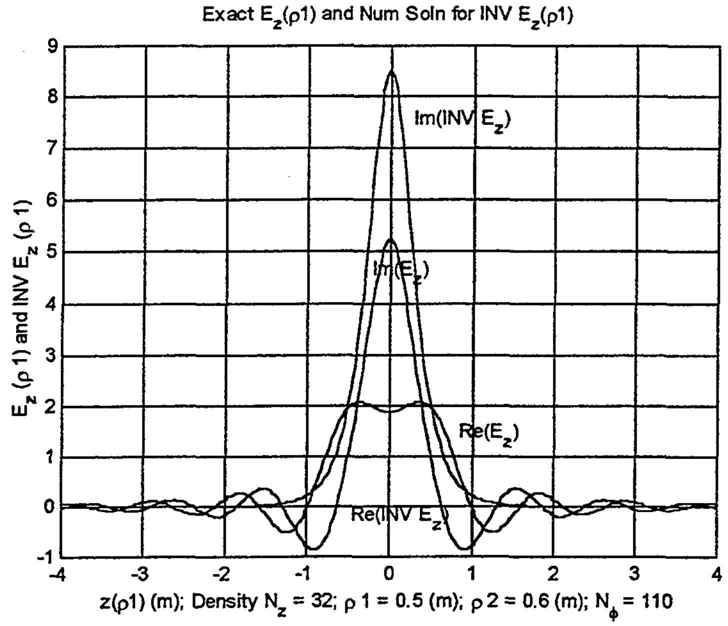


Figure 30. Back-Propagated and Exact Electric Field Components, Case 2

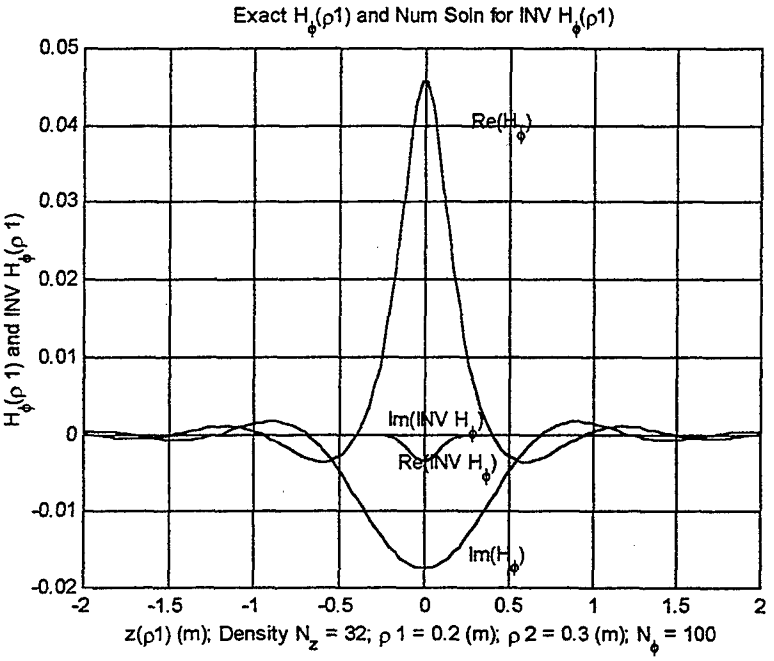


Figure 31. Back-Progagated and Exact Magnetic Field Components, Case 1

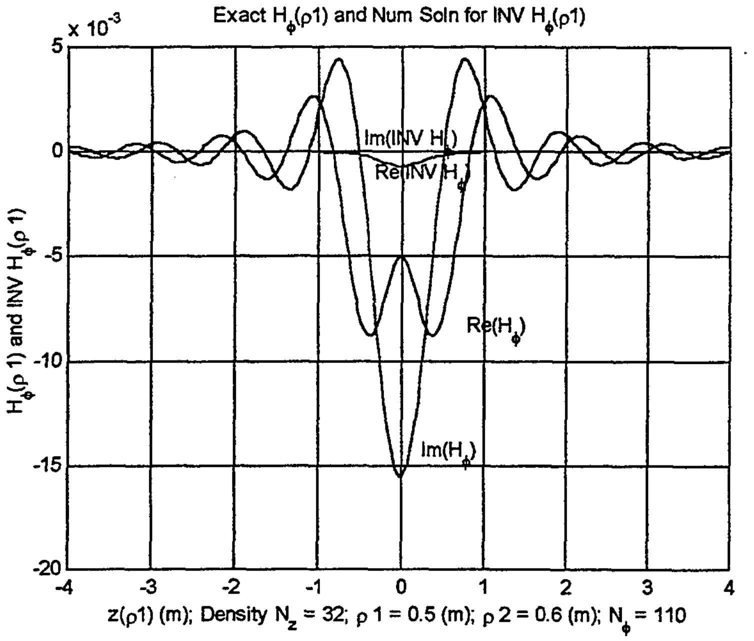


Figure 32. Back-Propagaged and Exact Magnetic Field Components, Case 2

The inverted fields seem to be composed of only the imaginary E-field and the Real H-field. To investigate the cause of this, the inverted fields at ρ_1 were propagated back to ρ_2 to validate the T matrix. As expected from a transform, inverse transform pair of operations, the inverted fields at ρ_1 recreated the calculated fields at ρ_2. Thus, using the T matrix, there are at least two field distributions on ρ_1 that create the fields at ρ_2. Upon further investigation, various combinations of the Real and Imaginary components of the exact fields on the ρ_1 cylindrical surface were propagated out to the ρ_2 cylindrical surface using the T matrix. Figures 33 through 36 show that the combination of the Imaginary E-field and Real H-field components of the ρ_1 field distribution recreated the desired fields at ρ_2. The two field components that formed the calculated ρ_2 field distribution are also the only non-zero fields produced when T^{-1} is applied to the exact ρ_2 fields. Figures 35 and 36 show that the combination of Real E-field and Imaginary H-field components of the ρ_1 field distributions seemed to produce zero fields at ρ_2. This led to the discovery that the real ρ_1 E-field and the imaginary ρ_1 H-field are complementary solutions to Maxwell's equations and are in the null space of the transform matrix T operator. [Ref. 10] Conversely this implies that there is a particular solution to Maxwell's equations that produces the fields at ρ_2. [Ref. 10]

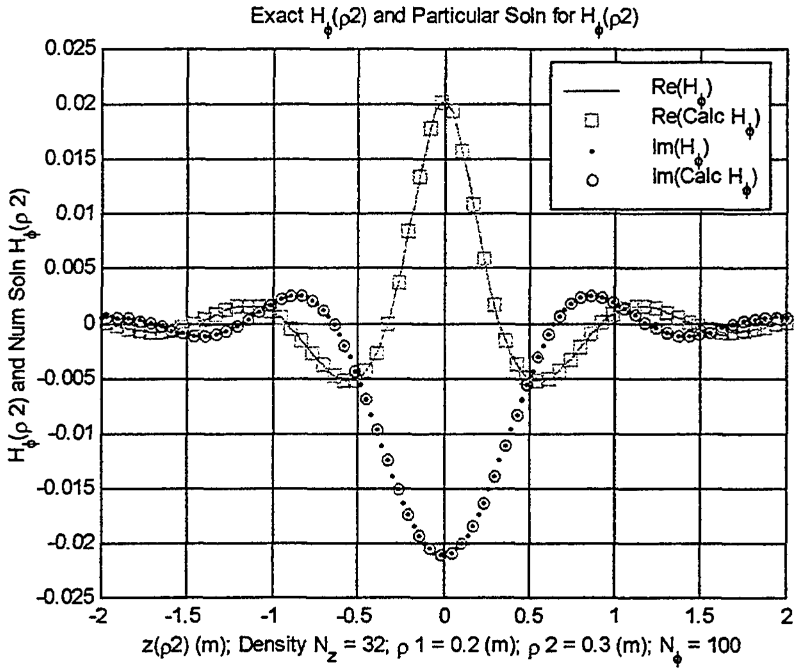


Figure 33. Exact and Particular Solution Component Calculated Magnetic Fields

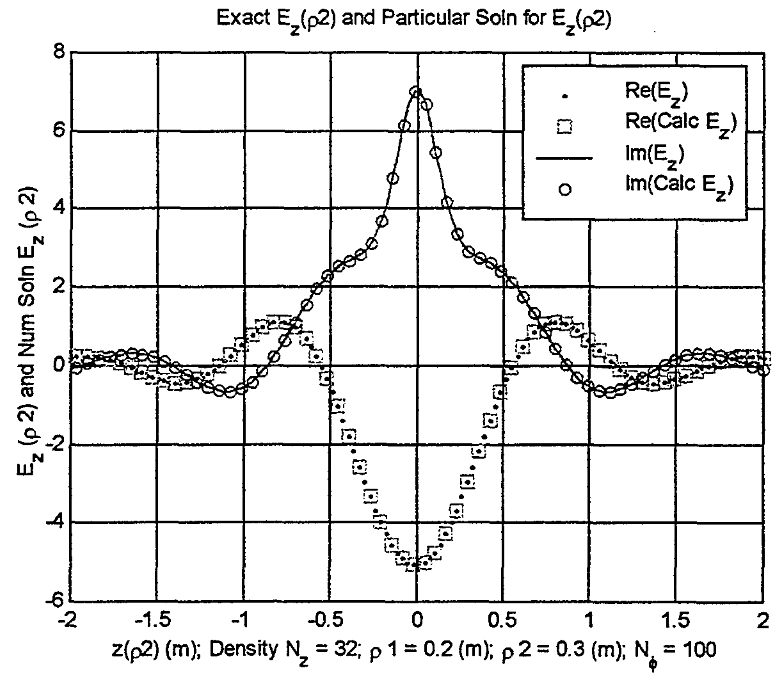


Figure 34. Exact and Particular Solution Component Calculated Electric Fields

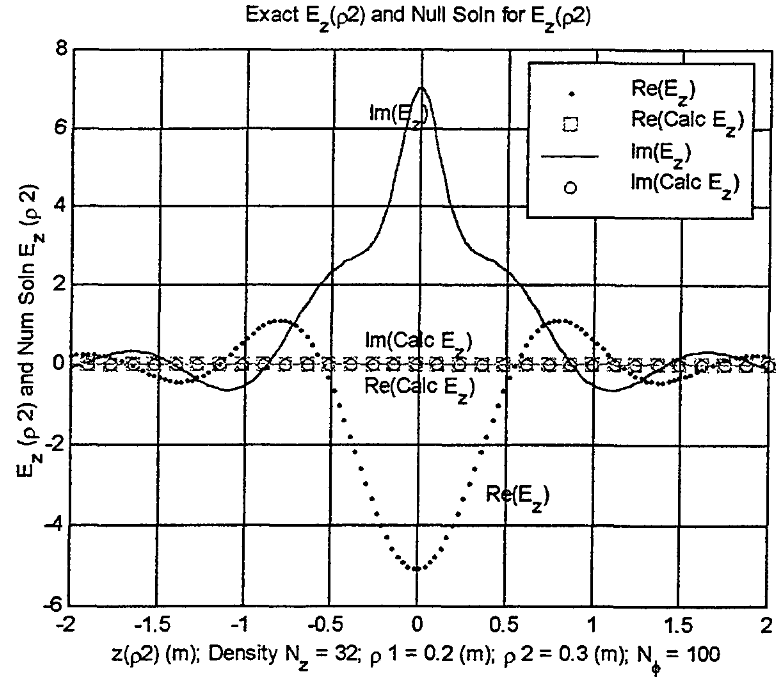


Figure 35. Exact and Null Solution Component Calculated Electric Fields

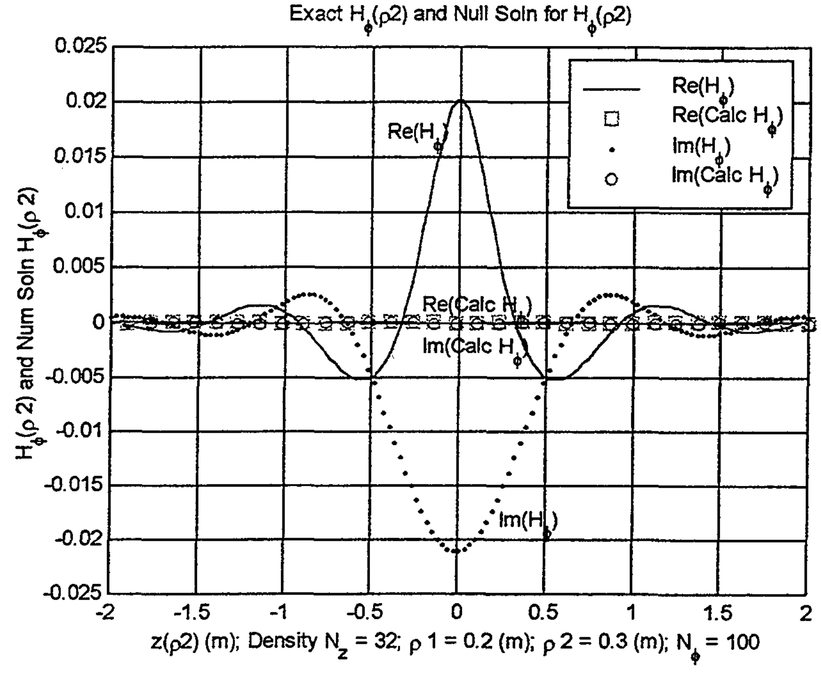


Figure 36. Exact and Null Solution Component Calculated Magnetic Fields

In order to better understand this, Maxwell's equations are examined for a strictly real current and for E and H-fields broken into real and imaginary components. While $\overline{J} = \overline{J_A} + j\overline{J_B}$ in general, the assumed dipole current distribution is strictly real, $\overline{J} = \overline{J_A}$.

$$\overline{E}_A = \overline{E}_{A2} + j\overline{E}_{A1} \qquad (90)$$

$$\overline{H}_A = \overline{H}_{A1} + j\overline{H}_{A2} \qquad (91)$$

The real and imaginary parts of $\overline{E}_A$ and $\overline{H}_A$ are indexed such that paired component solutions have identical indices. Substituting equations (90) and (91) into Maxwell's equations illustrates the respective field components relations.

$$\nabla \times \left(\overline{E}_{A2} + j\overline{E}_{A1}\right) = -j\omega\mu\left(\overline{H}_{A1} + j\overline{H}_{A2}\right) \qquad (92)$$

$$\nabla \times \left(\overline{H}_{A1} + j\overline{H}_{A2}\right) = j\omega\varepsilon\left(\overline{E}_{A2} + j\overline{E}_{A1}\right) + \overline{J}_A \qquad (93)$$

These two equations are then split into four by grouping real and imaginary terms.

$$\nabla \times \left(j\overline{E}_{A1}\right) = -j\omega\mu\left(\overline{H}_{A1}\right) \qquad (94)$$

$$\nabla \times \left(\overline{H}_{A1}\right) = j\omega\varepsilon\left(j\overline{E}_{A1}\right) + \overline{J}_A \qquad (95)$$

$$\nabla \times \left(\overline{E}_{A2}\right) = -j\omega\mu\left(j\overline{H}_{A2}\right) \qquad (96)$$

$$\nabla \times \left(j\overline{H}_{A2}\right) = j\omega\varepsilon\left(\overline{E}_{A2}\right) + 0 \qquad (97)$$

Thus $\overline{E}_{A1}$ and $\overline{H}_{A1}$ in equations (94) and (95) solve the particular solution for Maxwell's equations with a real source current $\overline{J}_A$. Similarly, $\overline{E}_{A2}$ and $\overline{H}_{A2}$ in equations (96) and (97) solve the complimentary or source free solution for Maxwell's equations. [Ref. 10]

The fields partitioned into particular and complimentary solutions of Maxwell's equations can then be transformed into equivalent electric and magnetic current distributions by the surface equivalence theorem. The equivelent surface currents are divided into those created by the particular fields and those created by the complimentary fields as shown in Figure 37.

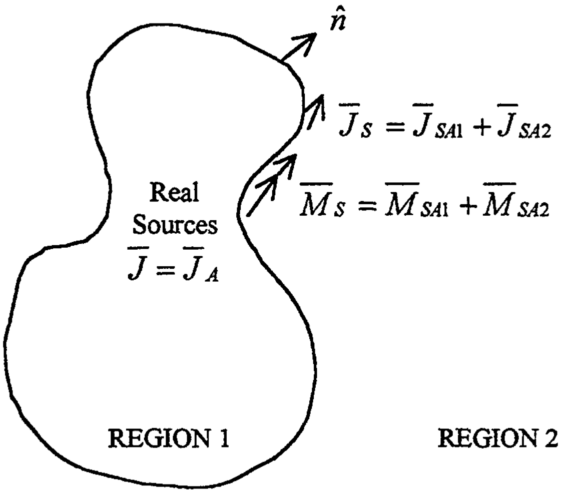


Figure 37. Equivalent Surface Currents for Partioned Field Solution

$$\overline{J}_{SA1}+\overline{J}_{SA2}=\hat{n}x\left(\overline{H}_{A1}+j\overline{H}_{A2}\right) \qquad (98)$$

$$\overline{M}_{SA1}+\overline{M}_{SA2}=\left(\overline{E}_{A2}+j\overline{E}_{A1}\right)x\hat{n} \qquad (99)$$

From the Surface Equivalence Theorem, one can now find the influence of just the particular solution fields and propagated fields in Region 2 outside the equivalent surface. A similar analysis can be done for the complimentary solution fields.

$$\overline{J}_{SA}\text{ and }\overline{M}_{SA} \quad \text{generate} \quad \begin{cases}\overline{E}=0 & \\ \overline{H}=0 & \text{Region1} \\ \overline{E}=\overline{E}_{A2}+j\overline{E}_{A1} & \\ \overline{H}=\overline{H}_{A1}+j\overline{H}_{A2} & \text{Region2}\end{cases}$$

$$\left.\begin{array}{l}\overline{J}_{SA1}=\hat{n}\times\overline{H}_{A1} \\ \overline{M}_{SA1}=j\overline{E}_{A1}\times\hat{n}\end{array}\right\} \quad \text{generate} \quad \begin{cases}\overline{E}=\overline{E}_{A2} & \\ \overline{H}=j\overline{H}_{A2} & \text{Region1} \\ \overline{E}=\overline{E}_{A2}+j\overline{E}_{A1} & \\ \overline{H}=\overline{H}_{A1}+j\overline{H}_{A2} & \text{Region2}\end{cases}$$

$$\left.\begin{array}{l}\overline{J}_{SA2}=\hat{n}\times\overline{H}_{A2} \\ \overline{M}_{SA2}=j\overline{E}_{A2}\times\hat{n}\end{array}\right\} \quad \text{generate} \quad \begin{cases}\overline{E}=-\overline{E}_{A2} & \\ \overline{H}=-j\overline{H}_{A2} & \text{Region1} \\ \overline{E}=0 & \\ \overline{H}=0 & \text{Region2}\end{cases}$$

[Ref. 10]

These results match the results from using the outward propagation transform matrix T and the particular and complimentary fields. The above equations suggest that using the transform matrix T on particular solution fields will give the complementary fields inside the surface. Put another way, use the T matrix on fields generated at ρ_2 to get some informaiton about fields at ρ_1. Figures 38 through 41 confirm this result. Figures 38 and 39 show the contributions to the fields inside ρ_1 from integrating the equivalent currents generated at ρ_1 by the particular solutions to Maxwell's equations. Inside ρ_1, the particular solution at ρ_1 (the Real H-field and the Imaginary E-field) create the null solutions inside ρ_1 (the Imaginary H-field and the Real E-field). Figures 40 and 41 show the contributions to the fields inside ρ_1 from integrating the equivalent currents generated at ρ_1 by the null solutions to Maxwell's equations. Inside ρ_1, the null solutions at ρ_1 create the inverse of the null solutions inside ρ_1. Thus, the total field from null and particular solutions will cancel inside ρ_1. Unfortunately, this method of back propagating a portion of the field gives only complimentary field information. Therefore, the inverse T method is still needed for back propagation of the particular solution.

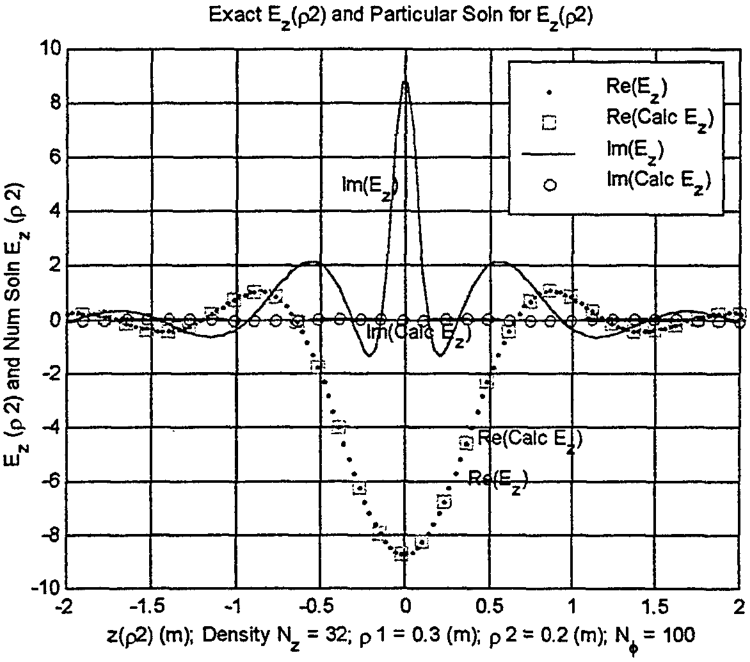


Figure 38. Exact and Integrated Particular Solution Electric Fields

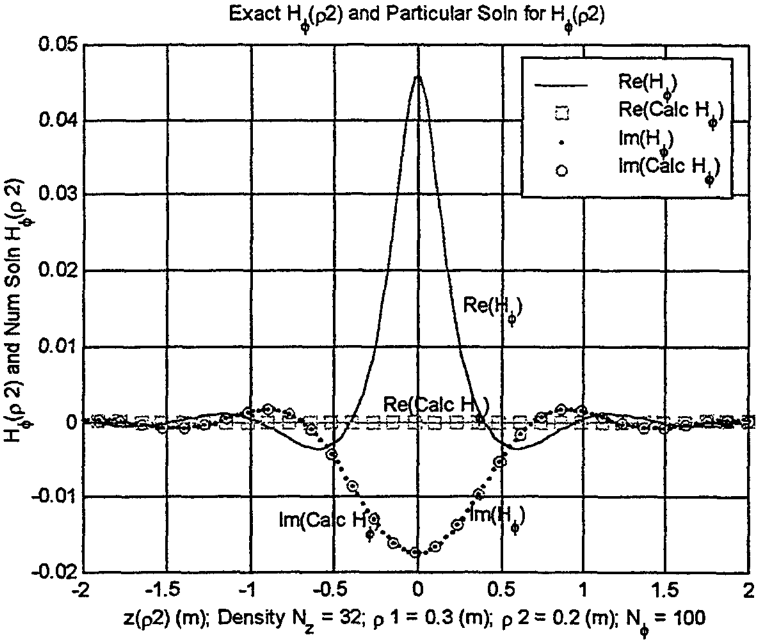


Figure 39. Exact and Integrated Particular Solution Magnetic Fields

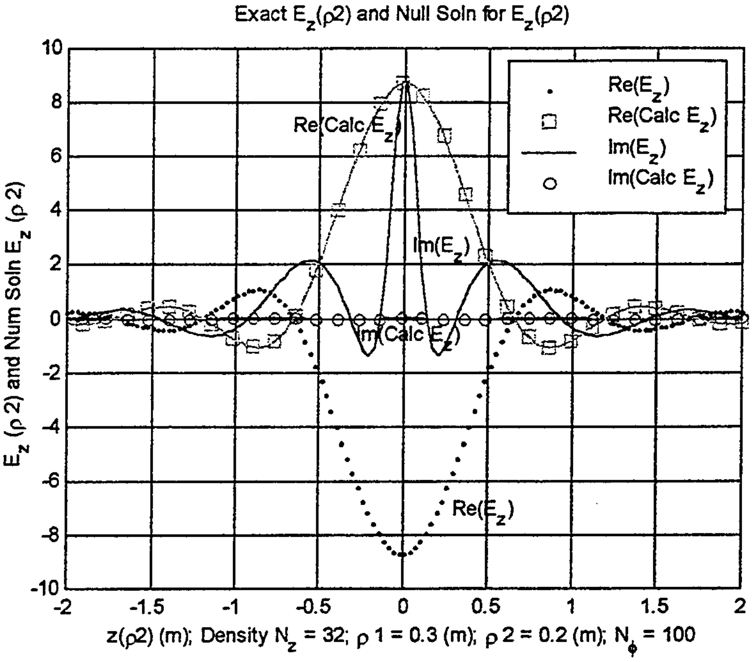


Figure 40. Exact and Integrated Null Solution Electric Fields

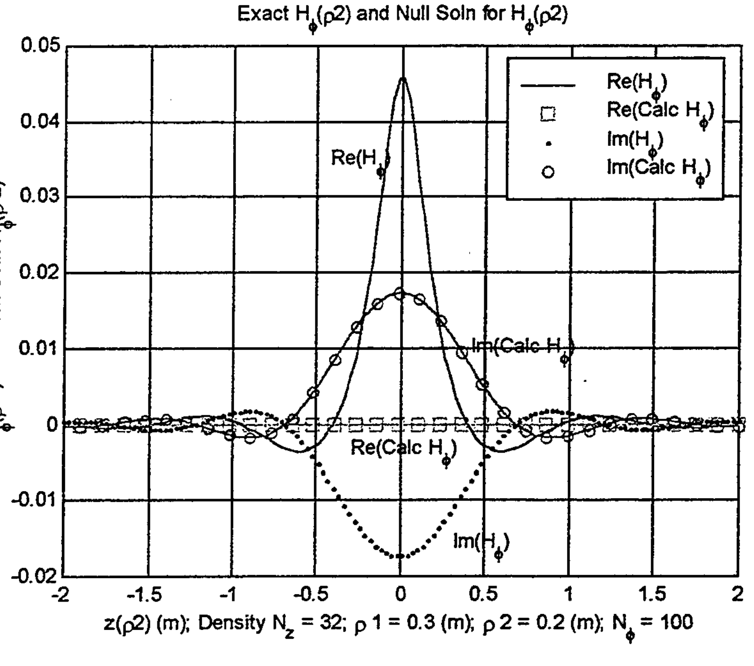


Figure 41. Exact and Integrated Null Solution Magnetic Fields

V. CONCLUSION

As the distance from a source increases, the ability to resolve rapid field variations close to the source degrades. Once in the far field, two sources can only be distinguished if they are greater than one half wavelength apart. If measurements are made on the surface of the object, the physical presence of the probe disrupts the fields it is attempting to sample. Moving a short distance away from the source into the near field provides a compromise between probe interference and resolution capability. The near field probe will no longer heavily disturb the fields and will be able to measure fields capable of providing better than one half wavelength localization of sources.

The techniques used to analyze back propagation do not give a perfect reconstruction of fields closer to the source. In fact, some components of the calculated inverse fields are essentially zero and provide no useful information about the location of the source whatsoever.

The reason that the inversion technique as used in this thesis does not give better answers is threefold. At the level of analyzing the transfer matrix structure, the matrices used are very ill-conditioned. In fact, as the matrices become more accurate—more z range with finer integrations, they should become "perfectly" ill-conditioned. They transform non-zero vectors to zero exactly. The physical nature of the fields innately creates ill-conditioned matrices. As more points and finer resolution are used to decrease error in the calculated fields at ρ_2, the transfer matrix thus produced becomes more and more ill-conditioned. Inverting the ill-conditioned matrix creates another ill-conditioned transfer matrix for determining the source fields at ρ_1. The matrix is so ill-conditioned that even a very small difference between the calculated and exact field at ρ_2 causes large error in the calculated inverse field.

At an intermediate level, an actual radiator with a cylindrical surface at ρ_1 may produce evanescent fields as well as the fields identical to those from a real source current located inside ρ_1. These evanescent fields would have a small but non-zero contribution to the fields measured at ρ_2. For equivalent currents at ρ_1, no actual evanescent fields exist. The equivalent currents are used to make numerical estimates of the fields at ρ_2—the calculated ρ_2 fields. When inverting the exact fields at ρ_2 to estimate the fields at ρ_1, the small differences between the calculated ρ_2 fields and the exact ρ_2 fields could be created by an evanescent field at ρ_1. Thus, this technique has the potential to produce non-physical evanescent fields as well as the actual fields on ρ_1.

At the highest level, there are the cases of non-radiating fields [Ref. 8 and 10]. The fields have a mathematical structure such that they can be large at ρ_1 and exactly zero at ρ_2 and at all points greater than ρ_2. These fields are the homogeneous solution to the forward transfer matrix and can be considered to be in the null space of the forward transfer matrix. These fields can exist mathematically, but discretizing the problem may cause the inverse transfer matrix to reconstruct the actual along with some part of a non-radiating or evanescent field.

Therefore, by its very nature the inverse source problem as handled in this technique is not only ill-conditioned, but is perfectly ill-conditioned, since null or very small fields at ρ_2 can exist due to large non-physical fields at ρ_{1s}. There will then be the actual inverse field plus any number of non-radiating and evanescent fields that can exist at ρ_1 and still give the measured exact fields at ρ_2.

The technique was chosen for its ability to be expanded to any axisymmetric problem, not just cylindrical symmetries, in a manner that did not exclusively depend on a separable geometry and specialized Hankel functions.

There are several areas for further study using this technique. The first is a study of modifications to the transfer function in order to eliminate its null space in the inversion process. The second is examining methods for decorrelating the terms in the T matrix to simplify inversion. The third is wavelet analysis of improved transfer matrices, specifically focusing on a wavelet's joint space/spatial scale localization abilities. The fourth is applying this technique to other axisymmetric fields and sources (such as a spherical radiator, conical radiator, and other sources found on a surface of revolution). Finally, this technique can be used to find the actual surface currents on metal radiators and scatterers.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

[1] Williams, E. G., "Supersonic Acoustic Intensity," *Journal of the Acoustical Society of America*, Vol 97, 1995, pp 121-127.

[2] Morgan, M. A., "EM Radiation Source Imaging using Superluminal Cylindrical Modes," In Preparation.

[3] Wawrzyniak, D. J. and Morgan, M. A., *Electromagnetic Imaging of Axisymmetric Scatterers*, Naval Postgraduate School, Monterey, CA, Master's Thesis, Jun 1997.

[4] Elliot, R. S., *Antenna Theory and Design*, 1981, Prentice-Hall, Englewood Cliffs, NJ, pp. 329-332.

[5] Balanis, C. A., *Advanced Engineering Electromagnetic*, 1989, Wiley & Sons, New York, NY, Chapter 6 pp 254-288.

[6] Morgan, M. A., Field Integrations for Axisymmetric Ring Currents, Unpublished Notes, dated 10 Dec 1998.

[7] Frued, Roland W., "A Look-Ahead Bareiss Algorithm for General Toeplitz Matrices," *Journal of Numerical Mathamatics*, Vol 68, 1994, pp 35-69.

[8] Jones, D. S., *Methods in Electromagnetic Wave Propagation*, 2nd Ed. 1994, Oxford, New York, NY, pp 568-570, 581-584, 607-608.

[9] Strang, G., *Linear Algebra and its Applications*, 3rd Ed 1988, Harcourt Brace & Co., New York, NY, Appendix A, pp 442-451.

[10] Morgan, M. A., Electric Source Field Pairs and Equivalent Surface Currents, Unpublished Notes, dated 20 Feb 1999.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A. METHOD B–MAXWELL'S CURL EQUATIONS APPLIED TWICE TO ORIGINAL VECTOR POTENTIALS AND METHOD C–SOLVING $\overline{\Delta \mathrm{E}_A}$ AND $\overline{\Delta \mathrm{H}}$ DIRECTLY FROM VECTOR POTENTIALS BY AN ALTERNATIVE METHOD

A. Method B: Maxwell's Curl Equations Applied Twice to Original Vector Potentials

In the previous section, $\overline{\Delta \mathrm{H}_F}$ and $\overline{\Delta E_A}$ were determined from $\overline{\Delta E_F}$ and $\overline{\Delta \mathrm{H}_A}$ respectively. In this method, $\overline{\Delta \mathrm{H}_F}$ and $\overline{\Delta E_A}$ are found directly without the intermediate step of completely determining $\overline{\Delta E_F}$ and $\overline{\Delta \mathrm{H}_A}$. Since

$$\overline{E_A} = \frac{1}{j\omega\varepsilon_\circ}\nabla\times\overline{H} = \frac{1}{j\omega\varepsilon_\circ}\left[\nabla\times\left[\frac{1}{\mu_\circ}\nabla\times\overline{A}\right]\right], \tag{100}$$

then $$\overline{E_A} = \frac{1}{j\omega\varepsilon_\circ}\left[\nabla\times\left[\frac{1}{\mu_\circ}\nabla\times\left[\frac{\mu_\circ}{4\pi}\iint_s \overline{J_s}\frac{e^{-jkR}}{R}ds'\right]\right]\right]. \tag{101}$$

The curl operations can both be brought inside the integral to give the following.

$$\overline{E_A} = \frac{1}{j\omega\varepsilon_\circ 4\pi}\iint_s\left[\nabla\times\left[\nabla\times\left(\overline{J_s}\frac{e^{-jkR}}{R}\right)\right]\right]ds' \tag{102}$$

Breaking the surface of the integration into many thin rings gives the following.

$$\overline{\Delta E_A} = \frac{\rho_1\Delta z_1}{j\omega\varepsilon_\circ 4\pi}\int_{-\pi}^{\pi}\left[\nabla\times\left[\nabla\times\left(J_z\hat{z}\frac{e^{-jkR}}{R}\right)\right]\right]d\phi_1 \tag{103}$$

In section II.B.2 "The Equivalent Electric Current Case and the Vector $\overline{A}$ Potential," it was found the following was found.

$$\nabla \times J_z \hat{z} \frac{e^{-jkR}}{R} = \left(\frac{-1}{R^3} + \frac{-jk}{R^2} \right) e^{-jkR} \left(\overline{R} \times \hat{z} \right) \tag{104}$$

Substituting this term yields the following.

$$\nabla \times \left[\left(\frac{-1}{R^3} + \frac{-jk}{R^2} \right) e^{-jkR} \left(\overline{R} \times \hat{z} \right) \right] = \left[\left(-\nabla \frac{e^{-jkR}}{R^3} - jk\nabla \frac{e^{-jkR}}{R^2} \right) \times \left(\overline{R} \times \hat{z} \right) \right.$$

$$\left. - \left(\frac{1}{R^3} + \frac{jk}{R^2} \right) e^{-jkR} \left(\nabla \times \left(\overline{R} \times \hat{z} \right) \right) \right] \tag{105}$$

This can be further expanded by using the $\nabla \frac{e^{-jkR}}{R^n}$ functions as follows.

$$\nabla \times \left[\nabla \times J_z \hat{z} \frac{e^{-jkR}}{R} \right] = \left(\frac{3}{R^5} + \frac{3jk}{R^4} - \frac{k^2}{R^3} \right) e^{-jkR} \left[\overline{R} \times \left[\overline{R} \times \hat{z} \right] \right]$$

$$- \left(\frac{1}{R^3} + \frac{jk}{R^2} \right) e^{-jkR} \left[\nabla \times \left[\overline{R} \times \hat{z} \right] \right] \tag{106}$$

Solving for $\left[\overline{R} \times \left[\overline{R} \times \hat{z} \right] \right]$ and $\left[\nabla \times \left[\overline{R} \times \hat{z} \right] \right]$ and substituting U$_n$, V$_n$, and W$_n$ functions then gives the final result.

$$\left(\Delta E_A \right)_\rho = \frac{\rho \Delta z \ \left(z_2 - z_1 \right) J_z}{j\omega \in_\circ 4\pi} \left[\rho_2 \left(3U_5 + 3jkU_4 - k^2 U_3 \right) \right.$$

$$\left. + \rho_1 \left(k^2 V_3 - 3jkV_4 - 3V_5 \ \right) \right] \tag{107}$$

$$\left(\Delta E_A \right)_z = \frac{{}_1\rho \Delta z \ J_z}{j\omega \in_\circ 4\pi} \left[2\left(U_3 + jkU_2 \right) + \left(\rho_1^2 + \rho_2^2 \right) \left(k^2 U_3 - 3jkU_4 - 3U_5 \right) \right.$$

$$\left. + \rho_1 \rho_2 \left(6V_5 + 6jkV_4 - 2k^2 V_3 \ \right) \right] \tag{108}$$

Note that while $(\Delta\mathrm{E}_A)_\rho$ is the same for Method A and Method B, $(\Delta\mathrm{E}_A)_z$ is somewhat different. Similarly, the following is true.

$$\overline{H_F} = -\frac{1}{j\omega\mu_\circ}\nabla\times\overline{E} = -\frac{1}{j\omega\mu_\circ}\left[\nabla\times\left[-\frac{1}{\varepsilon_\circ}\nabla\times\overline{F}\right]\right] \tag{109}$$

$$\overline{H_F} = -\frac{1}{j\omega\mu_\circ}\left[\nabla\times\left[-\frac{1}{\varepsilon_\circ}\nabla\times\left[\frac{\varepsilon_\circ}{4\pi}\iint_s \overline{M_s}\frac{e^{-jkR}}{R}ds'\right]\right]\right] \tag{110}$$

$$\overline{H_F} = -\frac{1}{j\omega\mu_\circ 4\pi}\iint_s\left[\nabla\times\left[\nabla\times\overline{M_s}\frac{e^{-jkR}}{R}\right]\right]ds' \tag{111}$$

Once again, breaking the cylindrical surface at ρ_1 into thin rings will give $\overline{\Delta\mathrm{H}}$.

$$\begin{aligned}\overline{\Delta\mathrm{H}_F} = \frac{M_\phi\rho\Delta z_1}{j\omega\mu_\circ 4\pi}\int_{-\pi}^{\pi}\Bigg[&\left(\frac{3}{R^5}+\frac{3jk}{R^4}-\frac{k^2}{R^3}\right)e^{-jkR}\left(\overline{R}\times\left(\overline{R}\times\left(\hat{\phi}\cos\phi_1-\hat{\rho}\sin\phi_1\right)\right)\right)\\ &-\left(\frac{1}{R^3}+\frac{jk}{R^2}\right)e^{-jkR}\left(\nabla\times\left(\overline{R}\times\left(\left(\hat{\phi}\cos\phi_1-\hat{\rho}\sin\phi_1\right)\right)\right)\right)\Bigg]d\phi_1\end{aligned} \tag{112}$$

$$\begin{aligned}(\Delta\mathrm{H}_F)_\phi = \frac{M_\phi\rho\Delta z_1}{j\omega\mu_\circ 4\pi}\Big[&2(V_3+jkV_2)+\left(\rho_1^2+\rho_2^2+(z_2-z_1)^2\right)\left(k^2V_3-3jkV_4-3V_5\right)\\ &+\rho_1\rho_2\left(3U_5+3jkU_4-k^2U_3+3W_5+3jkW_4-k^2W_3\right)\Big]\end{aligned} \tag{113}$$

This is the same $(\Delta\mathrm{H}_F)_\phi$ as determined in Method A.

B. Method C: Solving $\overline{\Delta E_A}$ and $\overline{\Delta H}$ Directly From Vector Potentials by an Alternative Method.

In the previous section $\overline{\Delta E_A}$ and $\overline{\Delta H_F}$ were solved directly from Maxwell's curl equations $\overline{E_A} = \frac{1}{j\omega\varepsilon_\circ}\nabla \times \overline{H_A}$ and $\overline{H_F} = \frac{1}{j\omega\mu_\circ}\nabla \times \overline{E_F}$. An alternative method is to substitute $\overline{H_A} = \frac{1}{\mu}\nabla \times \overline{A}$ and $\overline{E_F} = -\frac{1}{\varepsilon_\circ}\nabla \times \overline{F}$ and use Lorentz condition.

$$\overline{E_A} = -j\omega\overline{A} - \frac{j}{\omega\mu\varepsilon}\nabla\left(\nabla \bullet \overline{A}\right) \tag{114}$$

$$\overline{H_F} = -j\omega\overline{F} - \frac{j}{\omega\mu\varepsilon}\nabla\left(\nabla \bullet \overline{F}\right) \tag{115}$$

$$\overline{E_A} = -\frac{j\omega\mu}{4\pi}\iint_s J_z\hat{z}\frac{e^{-jkR}}{R}ds' - \frac{j}{\omega\varepsilon 4\pi}\iint_s\left[\nabla\left(\nabla \bullet J_z\hat{z}\frac{e^{-jkR}}{R}\right)\right]ds' \tag{116}$$

$$\overline{H_F} = -\frac{j\omega \in}{4\pi}\iint_s \overline{M_s}\frac{e^{-jkR}}{R}ds' - \frac{j}{\omega\mu 4\pi}\iint_s\left[\nabla\left(\nabla \bullet \overline{M_s}\frac{e^{-jkR}}{R}\right)\right]ds' \tag{117}$$

Expand the second terms of $\overline{E_A}$ and $\overline{H_F}$.

$$\nabla\left(\nabla \bullet J_z\hat{z}\frac{e^{-jkR}}{R}\right) = \nabla\left[\left(J_z\hat{z} \bullet \nabla\frac{e^{-jkR}}{R}\right) + \frac{e^{-jkR}}{R}\left(\nabla \bullet J_z\hat{z}\right)\right] \tag{118}$$

$$\nabla\left(\nabla \bullet J_z\hat{z}\frac{e^{-jkR}}{R}\right) = \nabla\left[\left(\frac{-1}{R^3} + \frac{-jk}{R^2}\right)e^{-jkR}\left(J_z\hat{z} \bullet \overline{R}\right)\right] \tag{119}$$

$$\nabla\left(\nabla\bullet J_z\hat{z}\frac{e^{-jkR}}{R}\right)=\left(z_2-z_1\right)\left(\frac{3}{R^5}+\frac{3jk}{R^4}-\frac{k^2}{R^3}\right)e^{-jkR}\overline{R}-\left(\frac{1}{R^3}+\frac{jk}{R^2}\right)e^{-jkR}\hat{z} \tag{120}$$

$$\nabla\left(\nabla\bullet\left(\overline{M_s}\frac{e^{-jkR}}{R}\right)\right)=\nabla\left[\left(\overline{M_s}\bullet\nabla\frac{e^{-jkR}}{R}\right)+\frac{e^{-jkR}}{R}\left(\nabla\bullet\overline{M_s}\right)\right] \tag{121}$$

$$\nabla\left(\nabla\bullet\left(\overline{M_s}\frac{e^{-jkR}}{R}\right)\right)=\nabla\left[\left(\frac{-1}{R^3}+\frac{-jk}{R^2}\right)e^{-jkR}\left(\overline{M_s}\bullet\overline{R}\right)\right] \tag{122}$$

$$\nabla\left(\nabla\bullet\left(\overline{M_s}\frac{e^{-jkR}}{R}\right)\right)=\rho_2\sin\phi_1\left(\frac{k^2}{R^3}-\frac{3jk}{R^4}-\frac{3}{R^5}\right)e^{-jkR}\overline{R}$$

$$+\sin\phi_1\left(\frac{1}{R^3}+\frac{jk}{R^2}\right)e^{-jkR}\hat{\rho}+\cos\phi_1\left(\frac{-1}{R^3}+\frac{-jk}{R^2}\right)e^{-jkR}\hat{\phi} \tag{123}$$

Break the cylindrical surface into thin rings to calculate $\overline{H_F}$ and $\overline{E_A}$ and substitute U_n, V_n, and W_n functions.

$\left(\Delta E_A\right)_\rho$ is the same as in the previous two methods.

$$\left(\Delta E_A\right)_z=\frac{\rho_1\Delta z J_z}{j4\pi}\left\{\omega\mu_o U_1\right.$$

$$\left.+\frac{1}{\omega\varepsilon_o}\left[\left(z_2-z_1\right)\left(3U_5+3jkU_4-k^2U_3\right)-U_3-jkU_2\right]\right\} \tag{124}$$

$$\left(\Delta \mathrm{H}_F\right)_\phi = \frac{M_\phi \rho_1 \Delta z}{j4\pi}\Big\{\omega\varepsilon_\circ V_1$$

$$+\frac{1}{\omega\varepsilon_\circ}\left[\rho_1\rho_2\left(3U_5 + 3jkU_4 - k^2U_3 - 3W_5 - 3jkW_4 + k^2W_3\right) - V_3 - jkV_2\right]\Big\} \qquad (125)$$

Note that while $\left(\Delta \mathrm{E}_A\right)_\rho$ is the same, $\left(\Delta \mathrm{E}_A\right)_z$ and $\left(\Delta H_F\right)_\phi$ are somewhat different in form when compared to the previous two methods.

APPENDIX B. COMPUTER SOURCE CODES DEVELOPED

```
% Test8a.m
% By M.A. Morgan 13 June 97
% Mod-8a on 30 May 98 substitutes ifft for fft and fft for ifft
% to correct kz polarity.

% Propagates exact H_phi, E_z and E_rho fields from a z-axis
% array of specified dipoles between rho1 and rho2 using
% cylindrical wave function transfer functions of complex FFT's.
% Fields are plotted on a user specified 2-D grid in rho and z.

% An N-element uniform array of identical dipole elements is the
% source.  A sequential phase shift on element input currents
% provides beam steering of the specified array.
% Dipole element currents have sinusoidal form of I(z)=Im sin(h-|z|)
% which has exact near-field integration results found in
% Jordan & Balmain pp 333-337 or Elliott pp 329-332.

clear all; eta=120*pi;
f=300;% f=input('Enter frequency (MHz): ');
L=input('Enter dipole element length L (meters): ');
h=L/2;
wl=300/f; k=2*pi/wl; k2=k*k; kh=k*h; Im=1; ckh=cos(kh);
N=input('Enter number of dipoles in array: ');
if N > 1,
   d=input('Enter array spacing (meters): ');
   phi=input('Enter far-field pointing angle (deg): ');
   zd=-(N-1)*d/2:d:(N-1)*d/2;    % Element positions
   alpha=-k*zd*cos(pi*phi/180); % Element phasing
   I=Im*exp(j*alpha);           % Input currents
else,
   zd=0; I=Im;
end

rho1=input('Enter rho1 (m) for field calculation grid: ');
rho2=input('Enter rho2 (m) for field calculation grid: ');
nr=input('Enter Number of rho-Points: ');
rho=linspace(rho1,rho2,nr);  % Row array

z1=input('Enter z1 (m) for field calculation grid: ');
z2=input('Enter z2 (m) for field calculation grid: ');
nz=input('Enter Number of z-Points (64, 128, 256, ... etc): ');
dz=(z2-z1)/(nz-1); zee=(z1:dz:z2)';             % Column array
% Wavenumber spectra
dkz=2*pi/(nz*dz); kz2=nz*dkz/2; kz1=-kz2+dkz;
kz=kz1:dkz:kz2; kzp=0:dkz:kz2; nkzp=length(kzp);
nsup=find(kzp <= k); nsub=find(kzp > k);
krsup=sqrt(k2-kzp(nsup).*kzp(nsup));
if ~isempty(nsub),
krsub=sqrt(kzp(nsub).*kzp(nsub)-k2); end

% Initializing Arrays
```

```
FE1=zeros(nkzp,1); FH1=FE1;  Hp=zeros(nz,nr); Ez=Hp; Er=Hp;
[Rho,Zee]=meshgrid(rho,zee); Rho2=Rho.*Rho;

for n=1:N            % Array element index
   Zn0=Zee-zd(n); Zn1=Zn0-h; Zn2=Zn0+h;
   R0=sqrt(Rho2 + Zn0.^2); R1=sqrt(Rho2 + Zn1.^2); R2=sqrt(Rho2 +
Zn2.^2);
   E0=2*exp(-j*k*R0); E1=exp(-j*k*R1); E2=exp(-j*k*R2);
   Hp(:,:)=Hp(:,:)+j*I(n)*(E1 + E2 - ckh*E0)./(4*pi*Rho);
   Ez(:,:)=Ez(:,:)-j*30*I(n)*(E1./R1 + E2./R2 - ckh*E0./R0);
   Er(:,:)=Er(:,:)+j*30*I(n)*...
      (Zn1.*E1./R1 + Zn2.*E2./R2 - ckh*Zn0.*E0./R0)./Rho;
end
svfl=input('Enter 1 to Save E and H Field Files: ');
if ~isempty(svfl), if svfl == 1,
      hdr=[f rho1 rho2 nr z1 z2 nz];
      save hdr.dat hdr /ascii
      dum=real(Hp); save hpr.dat dum -ascii
      dum=imag(Hp); save hpi.dat dum -ascii
      dum=real(Ez); save ezr.dat dum -ascii
      dum=imag(Ez); save ezi.dat dum -ascii
      dum=real(Er); save err.dat dum -ascii
      dum=imag(Er); save eri.dat dum -ascii
end; end
Fp1=zeros(nz,nr); Pp1=zeros(nz,1); Pp2=Pp1; Pp3=Pp1;
% FFT in z of each fixed rho column
% using ifft in z to correct polarity of k_z in FFT
Hpf=nz*ifft(Hp); Ezf=nz*ifft(Ez); Erf=nz*ifft(Er);

Pr=-pi*Rho.*real(Ez.*conj(Hp));  % Time-average rho-directed power/dz
Pz=+pi*Rho.*real(Er.*conj(Hp));  % Time-average z-directed power/dz

Prf=-Rho.*real(Ezf.*conj(Hpf))/2; % Power Spectral Densities (PSD)
Pzf=+Rho.*real(Erf.*conj(Hpf))/2;

while 1,
  disp('Program Options: ')
  disp('  0 ==> Stop Program')
  disp('  1 ==> H_phi 3-D Plot')
  disp('  2 ==> E_z 3-D Plot')
  disp('  3 ==> E_rho 3-D Plot')
  disp('  4 ==> P_rho 3-D Plot')
  disp('  5 ==> P_z 3-D Plot')
  disp('  6 ==> P-vector Plot')
  disp('  7 ==> Select Rho1 (Before Option 8)')
  disp('  8 ==> Fixed Propagation from Rho1 to Rho2')
  disp('  9 ==> Propagation Errors from Rho1 through Rho2')
  disp(' ')
nplot=input('Select Numbered Program Option: ');
if isempty(nplot)| nplot > 9 | nplot <=0, break; end

if nplot==1,
clf reset; surfl(Zee,Rho,abs(Hp),[0 0])
if N > 1,
   title(['|H_{phi}| for N=',int2str(N),' Array; d=',num2str(d),...
        ' m; L=',num2str(L),' m; phi=',num2str(phi),'deg; f=',...
         num2str(f),' MHz'])
```

```
else,
   title(['|H_{phi}| for L=',num2str(L),' m Dipole at f=',num2str(f),...
         ' MHz'])
end
xlabel('z (m)'); ylabel('rho (m)'); zlabel('|H_{phi}| (A/m)')
view(200,30); v=axis; axis([z1 z2 rho1 rho2 v(5) v(6)]); figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end
shading interp; view(90,-90); colorbar; figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

clf reset; surf(Zee,Rho,log10(abs(Hp)))
if N > 1,
   title(['|H_{phi}| for N=',int2str(N),' Array; d=',num2str(d),...
         ' m; L=',num2str(L),' m; phi=',num2str(phi),'deg; f=',...
         num2str(f),' MHz'])
else,
   title(['|H_{phi}| for L=',num2str(L),' m Dipole at f=',num2str(f),...
         ' MHz'])
end
xlabel('z (m)'); ylabel('rho (m)')
v=axis; view(90,-90); shading interp
axis([z1 z2 rho1 rho2 v(5) v(6)]); colorbar; figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

% Manual fftshift for 1-D fft array
Fp1(1:nkzp-2,:)=abs(Hpf(nkzp+1:nz,:));
Fp1(nkzp-1:nz,:)=abs(Hpf(1:nkzp,:));
clf reset; surfl(kz,rho,Fp1',[0 0])
if N > 1,
   title(['H_{phi} Spectrum for N=',int2str(N),' Array;
d=',num2str(d),...
         ' m; L=',num2str(L),' m; phi=',num2str(phi),'deg; f=',...
         num2str(f),' MHz'])
else,
   title(['H_{phi} Spectrum for L=',num2str(L),' m Dipole at f=',...
         num2str(f),' MHz'])
end
xlabel('k_z (rad/m)'); ylabel('rho (m)'); zlabel('|FFT [H_{phi}]|')
view(200,30); v=axis; axis([kz1 kz2 rho1 rho2 v(5) v(6)]); figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end
shading interp; view(90,-90); colorbar; figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

clf reset; surf(kz,rho,log10(Fp1'+.01))
if N > 1,
   title(['H_{phi} Spectrum for N=',int2str(N),' Array;
d=',num2str(d),...
         ' m; L=',num2str(L),' m; phi=',num2str(phi),'deg; f=',...
         num2str(f),' MHz'])
else,
   title(['H_{phi} Spectrum for L=',num2str(L),' m Dipole at f=',...
         num2str(f),' MHz'])
```

```
end
xlabel('k_z (rad/m)'); ylabel('rho (m)')
v=axis; view(90,-90); shading interp
axis([kz1 kz2 rho1 rho2 v(5) v(6)]); colorbar; figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end
end

if nplot==2,
clf reset; surfl(Zee,Rho,abs(Ez),[0 0])
if N > 1,
   title(['|E_z| for N=',int2str(N),' Array; d=',num2str(d),' m; L=',...
      num2str(L),' m; phi=',num2str(phi),'deg; f=',num2str(f),' MHz'])
else,
   title(['|E_z| for L=',num2str(L),' m Dipole at f=',num2str(f),'
MHz'])
end
xlabel('z (m)'); ylabel('rho (m)'); zlabel('|E_z| (V/m)')
view(200,30); v=axis; axis([z1 z2 rho1 rho2 v(5) v(6)]); figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end
shading interp; view(90,-90); colorbar; figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

clf reset; surf(Zee,Rho,log10(abs(Ez)))
if N > 1,
   title(['|E_z| for N=',int2str(N),' Array; d=',num2str(d),...
         ' m; L=',num2str(L),' m; phi=',num2str(phi),'deg; f=',...
         num2str(f),' MHz'])
else,
   title(['|E_z| for L=',num2str(L),' m Dipole at f=',num2str(f),...
         ' MHz'])
end
xlabel('z (m)'); ylabel('rho (m)')
v=axis; view(90,-90); shading interp
axis([z1 z2 rho1 rho2 v(5) v(6)]); colorbar; figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

% Manual fftshift for 1-D fft array
Fp1(1:nkzp-2,:)=abs(Ezf(nkzp+1:nz,:));
Fp1(nkzp-1:nz,:)=abs(Ezf(1:nkzp,:));
clf reset; surfl(kz,rho,Fp1',[0 0])
if N > 1,
   title(['E_z Spectrum for N=',int2str(N),' Array; d=',num2str(d),...
         ' m; L=',num2str(L),' m; phi=',num2str(phi),'deg; f=',...
         num2str(f),' MHz'])
else,
   title(['E_z Spectrum for L=',num2str(L),' m Dipole at f=',...
         num2str(f),' MHz'])
end
xlabel('k_z (rad/m)'); ylabel('rho (m)'); zlabel('|FFT [E_z]|')
view(200,30); v=axis; axis([kz1 kz2 rho1 rho2 v(5) v(6)]); figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end
shading interp; view(90,-90); colorbar; figure(1)
```

```
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

clf reset; surf(kz,rho,log10(Fp1'+.01))
if N > 1,
  title(['E_z Spectrum for N=',int2str(N),' Array; d=',num2str(d),...
      ' m; L=',num2str(L),' m; phi=',num2str(phi),'deg; f=',...
      num2str(f),' MHz'])
else,
  title(['E_z Spectrum for L=',num2str(L),' m Dipole at f=',...
      num2str(f),' MHz'])
end
xlabel('k_z (rad/m)'); ylabel('rho (m)')
v=axis; view(90,-90); shading interp
axis([kz1 kz2 rho1 rho2 v(5) v(6)]); colorbar; figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end
end

if nplot==3,
clf reset; surfl(Zee,Rho,abs(Er),[0 0])
if N > 1,
  title(['|E_{rho}| for N=',int2str(N),' Array; d=',num2str(d),...
      ' m; L=',num2str(L),' m; phi=',num2str(phi),...
      'deg; f=',num2str(f),' MHz']);
else,
  title(['|E_{rho}| for L=',num2str(L),' m Dipole at f=',num2str(f),...
      ' MHz'])
end
xlabel('z (m)'); ylabel('rho (m)'); zlabel('|E_{rho}| (V/m)')
view(200,30); v=axis; axis([z1 z2 rho1 rho2 v(5) v(6)]); figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end
shading interp; view(90,-90); colorbar; figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

clf reset; surf(Zee,Rho,log10(abs(Er)))
if N > 1,
  title(['|E_{rho}| for N=',int2str(N),' Array; d=',num2str(d),...
      ' m; L=',num2str(L),' m; phi=',num2str(phi),'deg; f=',...
      num2str(f),' MHz'])
else,
  title(['|E_{rho}| for L=',num2str(L),' m Dipole at f=',num2str(f),...
      ' MHz'])
end
xlabel('z (m)'); ylabel('rho (m)')
v=axis; view(90,-90); shading interp
axis([z1 z2 rho1 rho2 v(5) v(6)]); colorbar; figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

% Manual fftshift for 1-D fft array
Fp1(1:nkzp-2,:)=abs(Erf(nkzp+1:nz,:));
Fp1(nkzp-1:nz,:)=abs(Erf(1:nkzp,:));
clf reset; surfl(kz,rho,Fp1',[0 0])
if N > 1,
```

```
   title(['E_{rho} Spectrum for N=',int2str(N),' Array;
d=',num2str(d),...
          ' m; L=',num2str(L),' m; phi=',num2str(phi),'deg; f=',...
          num2str(f),' MHz'])
else,
   title(['E_{rho} Spectrum for L=',num2str(L),' m Dipole at f=',...
          num2str(f),' MHz'])
end
xlabel('k_z (rad/m)'); ylabel('rho (m)'); zlabel('|FFT [E_{rho}]|')
view(200,30); v=axis; axis([kz1 kz2 rho1 rho2 v(5) v(6)]); figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end
shading interp; view(90,-90); colorbar; figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

clf reset; surf(kz,rho,log10(Fp1'+.01))
if N > 1,
   title(['E_{rho} Spectrum for N=',int2str(N),' Array;
d=',num2str(d),...
          ' m; L=',num2str(L),' m; phi=',num2str(phi),'deg; f=',...
          num2str(f),' MHz'])
else,
   title(['E_{rho} Spectrum for L=',num2str(L),' m Dipole at f=',...
          num2str(f),' MHz'])
end
xlabel('k_z (rad/m)'); ylabel('rho (m)')
v=axis; view(90,-90); shading interp
axis([kz1 kz2 rho1 rho2 v(5) v(6)]); colorbar; figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end
end

if nplot==4,
clf reset; surfl(Zee,Rho,Pr,[0 0])
if N > 1,
   title(['P_{rho} for N=',int2str(N),' Array; d=',num2str(d),...
          ' m; L=',num2str(L),' m; phi=',num2str(phi),...
          'deg; f=',num2str(f),' MHz']);
else,
   title(['P_{rho} for L=',num2str(L),' m Dipole at f=',num2str(f),...
          ' MHz'])
end
xlabel('z (m)'); ylabel('rho (m)'); zlabel('P_{rho} (W/m)')
view(200,30); v=axis; axis([z1 z2 rho1 rho2 v(5) v(6)]); figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end
shading interp; view(90,-90); colorbar; figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

clf reset; surf(Zee,Rho,log10(abs(Pr)))
if N > 1,
   title(['P_{rho} for N=',int2str(N),' Array; d=',num2str(d),...
          ' m; L=',num2str(L),' m; phi=',num2str(phi),'deg; f=',...
          num2str(f),' MHz'])
else,
```

```
   title(['P_{rho} for L=',num2str(L),' m Dipole at f=',num2str(f),...
         ' MHz'])
end
xlabel('z (m)'); ylabel('rho (m)')
v=axis; view(90,-90); shading interp
axis([z1 z2 rho1 rho2 v(5) v(6)]); colorbar; figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

% Manual fftshift for 1-D fft array
Fp1(1:nkzp-2,:)=Prf(nkzp+1:nz,:); Fp1(nkzp-1:nz,:)=Prf(1:nkzp,:);
clf reset; surfl(kz,rho,Fp1',[0 0])
if N > 1,
   title(['P_r PSD for N=',int2str(N),' Array; d=',num2str(d),...
         ' m; L=',num2str(L),' m; phi=',num2str(phi),'deg; f=',...
         num2str(f),' MHz'])
else,
   title(['P_r PSD for L=',num2str(L),' m Dipole at f=',...
         num2str(f),' MHz'])
end
xlabel('k_z (rad/m)'); ylabel('rho (m)'); zlabel('P_r PSD')
view(200,30); v=axis; axis([kz1 kz2 rho1 rho2 v(5) v(6)]); figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end
shading interp; view(90,-90); colorbar; figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

clf reset; surf(kz,rho,log10(abs(Fp1)'+.01))
if N > 1,
   title(['P_r PSD for N=',int2str(N),' Array; d=',num2str(d),...
         ' m; L=',num2str(L),' m; phi=',num2str(phi),'deg; f=',...
         num2str(f),' MHz'])
else,
   title(['P_r PSD for L=',num2str(L),' m Dipole at f=',...
         num2str(f),' MHz'])
end
xlabel('k_z (rad/m)'); ylabel('rho (m)')
v=axis; view(90,-90); shading interp
axis([kz1 kz2 rho1 rho2 v(5) v(6)]); colorbar; figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end
end

if nplot==5,
clf reset; surfl(Zee,Rho,Pz,[0 0])
if N > 1,
   title(['P_z for N=',int2str(N),' Array; d=',num2str(d),...
         ' m; L=',num2str(L),' m; phi=',num2str(phi),...
         'deg; f=',num2str(f),' MHz']);
else,
   title(['P_z for L=',num2str(L),' m Dipole at f=',num2str(f),...
         ' MHz'])
end
xlabel('z (m)'); ylabel('rho (m)'); zlabel('P_z (W/m)')
view(200,30); v=axis; axis([z1 z2 rho1 rho2 v(5) v(6)]); figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
```

```
if ~isempty(hcpy), if hcpy == 1, print; end; end
shading interp; view(90,-90); colorbar; figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

clf reset; surf(Zee,Rho,log10(abs(Pz)))
if N > 1,
   title(['P_z for N=',int2str(N),' Array; d=',num2str(d),...
         ' m; L=',num2str(L),' m; phi=',num2str(phi),'deg; f=',...
         num2str(f),' MHz'])
else,
   title(['P_z for L=',num2str(L),' m Dipole at f=',num2str(f),...
         ' MHz'])
end
xlabel('z (m)'); ylabel('rho (m)')
v=axis; view(90,-90); shading interp
axis([z1 z2 rho1 rho2 v(5) v(6)]); colorbar; figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

% Manual fftshift for 1-D fft array
Fp1(1:nkzp-2,:)=Pzf(nkzp+1:nz,:); Fp1(nkzp-1:nz,:)=Pzf(1:nkzp,:);
clf reset; surfl(kz,rho,Fp1',[0 0])
if N > 1,
   title(['P_z PSD for N=',int2str(N),' Array; d=',num2str(d),...
         ' m; L=',num2str(L),' m; phi=',num2str(phi),'deg; f=',...
         num2str(f),' MHz'])
else,
   title(['P_z PSD for L=',num2str(L),' m Dipole at f=',...
         num2str(f),' MHz'])
end
xlabel('k_z (rad/m)'); ylabel('rho (m)'); zlabel('P_z PSD')
view(200,30); v=axis; axis([kz1 kz2 rho1 rho2 v(5) v(6)]); figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end
shading interp; view(90,-90); colorbar; figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

clf reset; surf(kz,rho,log10(abs(Fp1)'+.01))
if N > 1,
   title(['P_z PSD for N=',int2str(N),' Array; d=',num2str(d),...
         ' m; L=',num2str(L),' m; phi=',num2str(phi),'deg; f=',...
         num2str(f),' MHz'])
else,
   title(['P_z PSD for L=',num2str(L),' m Dipole at f=',...
         num2str(f),' MHz'])
end
xlabel('k_z (rad/m)'); ylabel('rho (m)')
v=axis; view(90,-90); shading interp
axis([kz1 kz2 rho1 rho2 v(5) v(6)]); colorbar; figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end
end

if nplot==6,
npwr=input('Enter inverse scaling factor for quiver plot: ')
```

```
Prmax=max(max(abs(Pr))); Pzmax=max(max(abs(Pz)));
Pmax=min(Prmax,Pzmax);
Pr0=sign(Pr).*(abs(Pr)/Pmax).^(1/npwr);
Pz0=sign(Pz).*(abs(Pz)/Pmax).^(1/npwr);
clf reset; quiver(Rho,Zee,Pr0,Pz0)
if N > 1,
   title(['P-vector for N=',int2str(N),' Array; d=',num2str(d),...
         ' m; L=',num2str(L),' m; phi=',num2str(phi),'deg; f=',...
         num2str(f),' MHz'])
else,
   title(['P-vector for L=',num2str(L),' m Dipole at f=',num2str(f),...
         ' MHz'])
end
xlabel('rho (m)'); ylabel('z (m)')
axis([rho1 rho2, z1 z2]); axis square; figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end
end

if nplot==7,
r1=input('Enter Rho1 radius to propagate from: ');

% Selecting closest grid radius
[rm, m1]=min(abs(rho-r1)); rp1=rho(m1);

% Forming denominator of radial transfer functions
FE1=zeros(nkzp,1); FH1=FE1;
[H, DH]=dhankel(0,krsup*rp1); FE1(nsup)=H; FH1(nsup)=DH;
if ~isempty(nsub),
[K, DK]=dmodbes(0,krsub*rp1); FE1(nsub)=K; FH1(nsub)=DK; end

% Energy Normalization for Optimal Estimators
EE1=FE1'*FE1/nkzp; EH1=FH1'*FH1/nkzp;

% Plotting Spatial and Spectral Fields at rho=rp1
clf reset
subplot(2,1,1),
plot(zee,abs(Ez(:,m1)),'g',zee,abs(eta*Hp(:,m1)),':r',...
   zee,abs(Er(:,m1)),'--c')
if N > 1,
   title(['Fields and Spectra; Array N=',int2str(N),';
d=',num2str(d),...
         'm; L=',num2str(L),'m; phi=',num2str(phi),'deg; f=',...
         num2str(f),'MHz; rho=',num2str(rp1),'m'])
else,
   title(['Fields and Spectra for L=',num2str(L),'m Dipole at f=',...
         num2str(f),'MHz; rho=',num2str(rp1),'m'])
end
xlabel('z (m)'); ylabel('V/m')
Pp1(1:nkzp-2)=abs(Ezf(nkzp+1:nz,m1)); Pp1(nkzp-
1:nz)=abs(Ezf(1:nkzp,m1));
Pp2(1:nkzp-2)=eta*abs(Hpf(nkzp+1:nz,m1)); % proper fftshift operation
Pp2(nkzp-1:nz)=eta*abs(Hpf(1:nkzp,m1));
Pp3(1:nkzp-2)=abs(Erf(nkzp+1:nz,m1)); Pp3(nkzp-
1:nz)=abs(Erf(1:nkzp,m1));
subplot(2,1,2), plot(kz,Pp1,'g',kz,Pp2,':r',kz,Pp3,'--c')
```

```
title('|E_z| (solid); eta*|H_{phi}| (short dash); |E_{rho}| (long
dash)')
xlabel('k_z (rad/m)'); ylabel('|FFT|'); figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

clf reset
subplot(2,1,1), plot(zee,Pr(:,m1),'g',zee,Pz(:,m1),'r:')
if N > 1,
   title(['Power & PSD; Array N=',int2str(N),'; d=',num2str(d),...
         'm; L=',num2str(L),'m; phi=',num2str(phi),'deg; f=',...
         num2str(f),'MHz; rho=',num2str(rp1),'m'])
else,
   title(['Fields and Spectra for L=',num2str(L),'m Dipole at f=',...
         num2str(f),'MHz; rho=',num2str(rp1),'m'])
end
xlabel('z (m)'); ylabel('W/m')
% proper fftshift operation
Pp1(1:nkzp-2)=Prf(nkzp+1:nz,m1); Pp1(nkzp-1:nz)=Prf(1:nkzp,m1);
Pp2(1:nkzp-2)=Pzf(nkzp+1:nz,m1); Pp2(nkzp-1:nz)=Pzf(1:nkzp,m1);
subplot(2,1,2), plot(kz,Pp1,'g',kz,Pp2,'r:')
title('Plots: P_r (solid)        P_z (dash)')
xlabel('k_z (rad/m)'); ylabel('PSD'); figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end
end

if nplot==8,
if isempty(r1), error('Enter Rho1 First'); end
r2=input('Enter Rho2 radius to propagate to: ');
% Selecting closest grid radius
[rm, m2]=min(abs(rho-r2)); rp2=rho(m2);

% Initialize Arrays
TEz=zeros(nz,1); THp=TEz; Pp1=TEz; Pp2=TEz;
FE2=zeros(nkzp,1); FH2=FE1;
% Forming numerator of radial transfer functions
[H, DH]=dhankel(0,krsup*rp2); FE2(nsup)=H; FH2(nsup)=DH;
if ~isempty(nsub),
[K, DK]=dmodbes(0,krsub*rp2); FE2(nsub)=K; FH2(nsub)=DK; end

while 2
CEz=input('Enter Optimal Estimator Constant for E_z (CEz < 0 to exit):
');
if CEz < 0, break; end
CHp=input('Enter Optimal Estimator Constant for H_phi & E_rho: ');
% Computing Riad's Optimal Deconvolution Propagation Transfer Functions
TEz(1:nkzp)=FE2.*conj(FE1)./(FE1.*conj(FE1) + CEz*EE1);
THp(1:nkzp)=FH2.*conj(FH1)./(FH1.*conj(FH1) + CHp*EH1);
% Filling in Negative k_z Spectrum Using Symmetry
TEz(nkzp+1:nz)=flipud(TEz(2:nkzp-1));
THp(nkzp+1:nz)=flipud(THp(2:nkzp-1));
% Plotting Magnitudes of Transfer Functions
% proper fftshift operation
Pp1(1:nkzp-2)=abs(TEz(nkzp+1:nz)); Pp1(nkzp-1:nz)=abs(TEz(1:nkzp));
Pp2(1:nkzp-2)=abs(THp(nkzp+1:nz)); Pp2(nkzp-1:nz)=abs(THp(1:nkzp));
clf reset; plot(kz,Pp1,'g',kz,Pp2,'r:')
```

```
title(['Transfer Functions: |TEz(k_z)| (solid);'...
      ' |THp(k_z)| (dash); Rho1=',num2str(rp1),...
      'm; Rho2=',num2str(rp2),'m; CEz=',num2str(CEz),...
      '; CHp=',num2str(CHp)])
xlabel('k_z (rad/m)'); ylabel('Magnitude'); figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

% Propagating Spectra and Inverse Transforming Fields
Ezf2=TEz.*Ezf(:,m1); Hpf2=THp.*Hpf(:,m1); Erf2=THp.*Erf(:,m1);
% Use fft as inverse to ifft for k_z polarity reversal in transforms
Ez2=fft(Ezf2)/nz; Hp2=fft(Hpf2)/nz; Er2=fft(Erf2)/nz;
Pr2=-pi*rp2.*real(Ez2.*conj(Hp2));  % rho-directed power/dz
Pz2=+pi*rp2.*real(Er2.*conj(Hp2));  % z-directed power/dz
Prf2=-rp2.*real(Ezf2.*conj(Hpf2))/2;% Power Spectral Densities
Pzf2=+rp2.*real(Erf2.*conj(Hpf2))/2;

% Computing RMS Errors in Spatial Fields at Rho2
DEz=Ez(:,m2)-Ez2; Ezerr=100*sqrt((DEz'*DEz)/(Ez(:,m2)'*Ez(:,m2)));
DHp=Hp(:,m2)-Hp2; Hperr=100*sqrt((DHp'*DHp)/(Hp(:,m2)'*Hp(:,m2)));
DEr=Er(:,m2)-Er2; Ererr=100*sqrt((DEr'*DEr)/(Er(:,m2)'*Er(:,m2)));
DPr=Pr(:,m2)-Pr2; Prerr=100*sqrt((DPr'*DPr)/(Pr(:,m2)'*Pr(:,m2)));
DPz=Pz(:,m2)-Pz2; Pzerr=100*sqrt((DPz'*DPz)/(Pz(:,m2)'*Pz(:,m2)));

% Plotting Spatial and Spectral Fields at rho=rp2
clf reset
subplot(2,1,1), plot(zee,abs(Ez(:,m2)),'g',zee,abs(Ez2),'r:')
title(['E_z at Rho=',num2str(rp2),'m Propagated from Rho=',...
      num2str(rp1),'m; CEz=',num2str(CEz),'; RMS Error=',...
      num2str(Ezerr),'%'])
xlabel('z (m)'); ylabel('V/m')
Pp1(1:nkzp-2)=abs(Ezf(nkzp+1:nz,m2)); Pp1(nkzp-
1:nz)=abs(Ezf(1:nkzp,m2));
Pp2(1:nkzp-2)=abs(Ezf2(nkzp+1:nz));    Pp2(nkzp-1:nz)=abs(Ezf2(1:nkzp));
subplot(2,1,2), plot(kz,Pp1,'g',kz,Pp2,'r:')
title('Plot Key:   Exact (solid)       Propagated (dash)')
xlabel('k_z (rad/m)'); ylabel('|FFT|'); figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

clf reset
subplot(2,1,1), plot(zee,abs(Hp(:,m2)),'g',zee,abs(Hp2),'r:')
title(['H_{phi} at Rho=',num2str(rp2),'m Propagated from Rho=',...
      num2str(rp1),'m; CHp=',num2str(CHp),'; RMS Error=',...
      num2str(Hperr),'%'])
xlabel('z (m)'); ylabel('V/m')
Pp1(1:nkzp-2)=abs(Hpf(nkzp+1:nz,m2)); Pp1(nkzp-
1:nz)=abs(Hpf(1:nkzp,m2));
Pp2(1:nkzp-2)=abs(Hpf2(nkzp+1:nz));    Pp2(nkzp-1:nz)=abs(Hpf2(1:nkzp));
subplot(2,1,2), plot(kz,Pp1,'g',kz,Pp2,'r:')
title('Plot Key:   Exact (solid)       Propagated (dash)')
xlabel('k_z (rad/m)'); ylabel('|FFT|'); figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

clf reset
subplot(2,1,1), plot(zee,abs(Er(:,m2)),'g',zee,abs(Er2),'r:')
```

```
title(['E_{rho} at Rho=',num2str(rp2),'m Propagated from Rho=',...
      num2str(rp1),'m; CEr=CHp=',num2str(CHp),'; RMS Error=',...
      num2str(Ererr),'%'])
xlabel('z (m)'); ylabel('V/m')
Pp1(1:nkzp-2)=abs(Erf(nkzp+1:nz,m2)); Pp1(nkzp-
1:nz)=abs(Erf(1:nkzp,m2));
Pp2(1:nkzp-2)=abs(Erf2(nkzp+1:nz));   Pp2(nkzp-1:nz)=abs(Erf2(1:nkzp));
subplot(2,1,2), plot(kz,Pp1,'g',kz,Pp2,'r:')
title('Plot Key:   Exact (solid)        Propagated (dash)')
xlabel('k_z (rad/m)'); ylabel('|FFT|'); figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

clf reset
subplot(2,1,1), plot(zee,Pr(:,m2),'g',zee,Pr2,'r:')
title(['P_{rho} at Rho=',num2str(rp2),'m Propagated from Rho=',...
      num2str(rp1),'m; CEz=',num2str(CEz),'; CHp=',num2str(CHp),...
      '; RMS Error=',num2str(Prerr),'%'])
xlabel('z (m)'); ylabel('W/m')
Pp1(1:nkzp-2)=Prf(nkzp+1:nz,m2); Pp1(nkzp-1:nz)=Prf(1:nkzp,m2);
Pp2(1:nkzp-2)=Prf2(nkzp+1:nz);   Pp2(nkzp-1:nz)=Prf2(1:nkzp);
subplot(2,1,2), plot(kz,Pp1,'g',kz,Pp2,'r:')
title('Plot Key:   Exact (solid)        Propagated (dash)')
xlabel('k_z (rad/m)'); ylabel('P_{rho} PSD'); figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

clf reset
subplot(2,1,1), plot(zee,Pz(:,m2),'g',zee,Pz2,'r:')
title(['P_z at Rho=',num2str(rp2),'m Propagated from Rho=',...
      num2str(rp1),'m; CEr=CHp=',num2str(CHp),...
      '; RMS Error=',num2str(Pzerr),'%'])
xlabel('z (m)'); ylabel('W/m')
Pp1(1:nkzp-2)=Pzf(nkzp+1:nz,m2); Pp1(nkzp-1:nz)=Pzf(1:nkzp,m2);
Pp2(1:nkzp-2)=Pzf2(nkzp+1:nz);   Pp2(nkzp-1:nz)=Pzf2(1:nkzp);
subplot(2,1,2), plot(kz,Pp1,'g',kz,Pp2,'r:')
title('Plot Key:   Exact (solid)        Propagated (dash)')
xlabel('k_z (rad/m)'); ylabel('P_z PSD'); figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end
end; end

if nplot==9,
if isempty(r1), error('Enter Rho1 First'); end
r2=input('Enter Rho2 for Endpoint of Propagation Error Calculation: ');
if r2==r1, error('Enter Rho2 Different Then Rho1'); end
% Selecting closest grid radius
[rm, m2]=min(abs(rho-r2));
if m2 > m1, M=m1:m2; else, M=m2:m1; end
rp=rho(M); nrp=length(rp);
FE2=zeros(nkzp,nrp); FH2=FE2; TEz=zeros(nz,nrp); THp=TEz;
% Forming denominator of radial transfer functions
for m=1:nrp;
[H, DH]=dhankel(0,krsup*rp(m));
FE2(1:nkzsup,m)=H; FH2(1:nkzsup,m)=DH;
if ~isempty(krsub),    % subluminal spectra
   [K, DK]=dmodbes(0,krsub*rp(m));
```

```
   FE2(nkzsup+1:nkzp,m)=K;
   FH2(nkzsup+1:nkzp,m)=DK;
end; end
FEE=FE1*ones(1,nrp); FHH=FH1*ones(1,nrp); Pp1=zeros(nz,nrp); Pp2=Pp1;

while 2
CEz=input('Enter Optimal Estimator Constant for E_z (CEz < 0 to exit):
');
if CEz < 0, break; end
CHp=input('Enter Optimal Estimator Constant for H_phi & E_rho: ');
% Computing Riad's Optimal Deconvolution Propagation Transfer Functions
TEz(1:nkzp,:)=FE2.*conj(FEE)./(FEE.*conj(FEE) + CEz*EE1);
THp(1:nkzp,:)=FH2.*conj(FHH)./(FHH.*conj(FHH) + CHp*EH1);
% Filling in Negative k_z Spectrum Using Symmetry
TEz(nkzp+1:nz,:)=flipud(TEz(2:nkzp-1,:));
THp(nkzp+1:nz,:)=flipud(THp(2:nkzp-1,:));
% Plotting Magnitudes of Transfer Functions
% proper fftshift operation
Pp1(1:nkzp-2,:)=abs(TEz(nkzp+1:nz,:));
Pp1(nkzp-1:nz,:)=abs(TEz(1:nkzp,:));
Pp2(1:nkzp-2,:)=abs(THp(nkzp+1:nz,:));
Pp2(nkzp-1:nz,:)=abs(THp(1:nkzp,:));

clf reset; mesh(rp,kz,Pp1)
title(['Deconvolution Function: |TEz(kz,rho)| with '...
  'CEz=',num2str(CEz),'; Propagate from Rho1=',num2str(rp1),'m'])
ylabel('k_z (rad/m)'); xlabel('rho (m)'); view(135,40); figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

clf reset; mesh(rp,kz,Pp2)
title(['Deconvolution Transfer Function: |THp(kz,rho)| for '...
  'CHp=',num2str(CHp),'; Propagate from Rho1=',num2str(rp1),'m'])
ylabel('k_z (rad/m)'); zlabel('rho (m)'); view(135,40); figure(1)
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

% Propagating Spectra and Inverse Transforming Fields
Ezf2=TEz.*(Ezf(:,m1)*ones(1,nrp));
Hpf2=THp.*(Hpf(:,m1)*ones(1,nrp));
Erf2=THp.*(Erf(:,m1)*ones(1,nrp));
Ez2=fft(Ezf2)/nz; Hp2=fft(Hpf2)/nz; Er2=fft(Erf2)/nz;
Pr2=-pi*(ones(nz,1)*rp).*real(Ez2.*conj(Hp2));  % rho-directed power/dz
Pz2=+pi*(ones(nz,1)*rp).*real(Er2.*conj(Hp2));  % z-directed power/dz

% Computing RMS Errors in Spatial Fields at Rho2
DEz=Ez(:,M)-Ez2; DHp=Hp(:,M)-Hp2; DEr=Er(:,M)-Er2;
DPr=Pr(:,M)-Pr2; DPz=Pz(:,M)-Pz2;
for m=1:nrp;
m2=M(m);
Ezerr(m)=100*sqrt((DEz(:,m)'*DEz(:,m))/(Ez(:,m2)'*Ez(:,m2)));
Hperr(m)=100*sqrt((DHp(:,m)'*DHp(:,m))/(Hp(:,m2)'*Hp(:,m2)));
Ererr(m)=100*sqrt((DEr(:,m)'*DEr(:,m))/(Er(:,m2)'*Er(:,m2)));
Prerr(m)=100*sqrt((DPr(:,m)'*DPr(:,m))/(Pr(:,m2)'*Pr(:,m2)));
Pzerr(m)=100*sqrt((DPz(:,m)'*DPz(:,m))/(Pz(:,m2)'*Pz(:,m2)));
end
```

```
clf reset; plot(rp,Ezerr,'g',rp,Hperr,'r:',rp,Ererr,'c--')
xlabel('Rho (m)'); ylabel('RMS Error (%)');
v= axis; v(3)=0; axis(v); figure(1)
title(['Error in E_z (solid), H_{phi} (short dash) and E_{rho}'...
      ' (long dash) Propagated from Rho1=',num2str(rp1),...
      ';CEz=',num2str(CEz),'; CHp=',num2str(CHp)])
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

clf reset; plot(rp,Prerr,'g',rp,Pzerr,'r:')
xlabel('Rho (m)'); ylabel('RMS Error (%)')
v= axis; v(3)=0; axis(v); figure(1)
title(['Error in P_r (solid) and P_z (dash) Propagated from Rho1=',...
      num2str(rp1),';CEz=',num2str(CEz),'; CHp=',num2str(CHp)])
hcpy=input('Enter 1 for Hard Copy: ');
if ~isempty(hcpy), if hcpy == 1, print; end; end

end; end; end
```

```
% EHFLDTST9.M tests dRhoEHFld function routine by comparing computation
% for sinusoidal I(z)=Im sin k(h-|z|) at selected cylindrical
% radius. (Dipole at the origin with sinusoidal Current Distribution).
% Using exact integration results for this assumed current
% given in Jordan & Balmain pp 333-337 and Elliot pp 329-332 for
% the Exact Fields at radius 1 and radius 2(> radius 1) and the
% Computed Ez and Hp fields at radius 2 given the fields at radius 1.
% Adapted from EHFLDTST.M by M.A. Morgan 6/6/97
% by D.G. Steenman 12/28/98
% Mod 9 of EHFLDTST: Calls dRhoEHFld4 which uses different field
% calculations.
% Calls dRhoEHFld4 (which calls intphi3 and phisquare)

clear all
L=input('Enter dipole length L (meters): '); h=L/2;
f=input('Enter frequency (MHz): ');
wl=300/f; k=2*pi/wl; kh=k*h; Im=1; ckh=cos(kh);
Invk = 1/k;
disp(['k*h = ',num2str(kh)]);
disp(['1/k = ',num2str(Invk)]);

while 1


zmax=input('Enter zmax (m) for z=[-zmax ... zmax] plots (0 to stop): ');
if ~isempty(zmax), if zmax == 0, break; end; end
M=input('Enter # z-points for [-zmax ... zmax]: ');
dz=2*zmax/(M-1); z=(-zmax:dz:zmax);

rho1=input('Enter rho1 (m) for plot (0 to stop): ');
if ~isempty(rho1), if rho1 == 0, break; end; end
rho2min=input('Enter rho2 min (m) for [rho2min ... rho2max] (0 to stop):
');
if ~isempty(rho2min), if rho2min == 0, break; end; end
rho2max=input('Enter rho2 max (m) for [rho2min ... rho2max] (0 to stop):
');
if ~isempty(rho2max), if rho2max == 0, break; end; end
Nrho = input(' Enter # rho2-points for [rho2min ... rho2max]: ');

dr=(rho2max-rho2min)/(Nrho-1); DRHO=(rho2min:dr:rho2max);

NpCalc = ceil(pi*2*rho1/dz)
Nplow=input('Enter Min number of Phi segments around a thin ring (radius
1) : ');

if NpCalc>Nplow,
   Np=NpCalc
else,
   Np=Nplow
end

% Exact Field Calculation at Radius 1
z1=z-h; z2=z+h;
r=sqrt(rho1*rho1+z.*z); E=exp(-j*k*r);
R1=sqrt(rho1*rho1+z1.*z1); E1=exp(-j*k*R1);
R2=sqrt(rho1*rho1+z2.*z2); E2=exp(-j*k*R2);
```

```
Ez1=-j*30*Im*(E1./R1+E2./R2-2*ckh*E./r);
Hp1=(j*Im/(4*pi))*(E1+E2-2*ckh*E)/rho1;

Jz=Hp1; Mp=Ez1;

% Set Up Loop to do sets of rho2 Field Calculations
EZRHO2 = zeros(length(z),length(DRHO));
HPRHO2 = zeros(length(z),length(DRHO));
EZRHO2N = zeros(length(z),length(DRHO));
HPRHO2N = zeros(length(z),length(DRHO));
EZRHO2NT = zeros(length(z),length(DRHO));
HPRHO2NT = zeros(length(z),length(DRHO));

for a = 1:(length(DRHO)),
   rho2 = DRHO(a);

% Exact Field Calculation at Radius 2
rrho2=sqrt(rho2*rho2+z.*z); Erho2=exp(-j*k*rrho2);
R1rho2=sqrt(rho2*rho2+z1.*z1); E1rho2=exp(-j*k*R1rho2);
R2rho2=sqrt(rho2*rho2+z2.*z2); E2rho2=exp(-j*k*R2rho2);
Ez2=-j*30*Im*(E1rho2./R1rho2+E2rho2./R2rho2-2*ckh*Erho2./rrho2);
Hp2=(j*Im/(4*pi))*(E1rho2+E2rho2-2*ckh*Erho2)/rho2;
EZRHO2(:,a)=Ez2.';
HPRHO2(:,a)=Hp2.';

% Numerical Integration using dRhoEHFld function for Radius 2
[Ez2N, Ez2NT, Hp2N, Hp2NT]=dREHF4(Np,k,rho1,rho2,z,z,z,z,1,1,Jz,Mp);
EZRHO2N(:,a)=Ez2N.';
HPRHO2N(:,a)=Hp2N.';
EZRHO2NT(:,a)=Ez2NT.';
HPRHO2NT(:,a)=Hp2NT.';

%clf reset;
figure(2*a-1)
plot(z,real(Ez2),'g',z,real(Ez2N),':r',z,real(Ez2NT),'-.y',...
     z,imag(Ez2),'g',z,imag(Ez2N),':r',z,imag(Ez2NT),'-.y',...
     z,abs(Ez2),'g',z,abs(Ez2N),':r',z,abs(Ez2NT),'-.y')
title(['E_z: Exact (solid); Integ (dot); NullTest (dashdot) at
{\rho}2=',num2str(rho2),...
      ' m from {\rho}1= ',num2str(rho1),' m for L=',num2str(L),' m;
N_z=',num2str(M),...
      '; N_{\phi}=',int2str(Np),'; f=',num2str(f),' MHz']);
xlabel(['z (m), Density N_z = ',num2str(M/(2*zmax))]); ylabel('Real and
Imag E_z (V/m)');
grid; orient landscape;
%hcpy=input('Print Hard Copy ? (Y/N): ','s');
%if ~isempty(hcpy), if hcpy=='Y'|hcpy=='y',orient landscape; print; end;
end

%clf reset;
figure(2*a)
plot(z,real(Hp2),'g',z,real(Hp2N),':r',z,real(Hp2NT),'-.y',...
     z,imag(Hp2),'g',z,imag(Hp2N),':r',z,imag(Hp2NT),'-.y',...
     z,abs(Hp2),'g',z,abs(Hp2N),':r',z,abs(Hp2NT),'-.y')
title(['H_{phi}: Exact (solid); Integ (dot); NullTest (dashdot) at
{\rho}2=',num2str(rho2),...
```

```
      ' m from {\rho}1= ',num2str(rho1),' m for L=',num2str(L),' m;
N_z=',num2str(M),...
      '; N_{\phi}=',int2str(Np),'; f=',num2str(f),' MHz']);
xlabel(['z (m), Density N_z = ',num2str(M/(2*zmax))]); ylabel('Real and
Imag H_{phi} (A/m)');
grid; orient landscape;

%hcpy=input('Print Hard Copy ? (Y/N): ','s');
%if ~isempty(hcpy), if hcpy=='Y'|hcpy=='y', orient landscape; print;
end; end

end

 d = 2*length(DRHO);
% d = 0;
[NRHO2G,ZEEG]=meshgrid(DRHO./rho1,z);
RErMagH = ((abs(HPRHO2-HPRHO2N)))./((abs(HPRHO2)));
RErMagE = ((abs(EZRHO2-EZRHO2N)))./((abs(EZRHO2)));

LSErMagH = sqrt((sum((abs(HPRHO2-
HPRHO2N)).^2))./(sum((abs(HPRHO2)).^2)))*100;
LSErMagE = sqrt((sum((abs(EZRHO2-
EZRHO2N)).^2))./(sum((abs(EZRHO2)).^2)))*100;

RErMagHNT = ((abs(HPRHO2-HPRHO2NT)))./((abs(HPRHO2)));
RErMagENT = ((abs(EZRHO2-EZRHO2NT)))./((abs(EZRHO2)));

LSErMagHNT = sqrt((sum((abs(HPRHO2-
HPRHO2NT)).^2))./(sum((abs(HPRHO2)).^2)))*100;
LSErMagENT = sqrt((sum((abs(EZRHO2-
EZRHO2NT)).^2))./(sum((abs(EZRHO2)).^2)))*100;


figure(d+1)
surf(NRHO2G,ZEEG,RErMagH)
title([' Relative Error in Exact to Num Soln for Mag(H_{\phi} ({\rho}
2)); {\rho} 1 = ',num2str(rho1),' (m) '])
xlabel([' {\rho} 2/{\rho} 1 ; N_{\rho}= ',num2str(Np)]);
ylabel([' z (m) ; N_z = ',num2str(M),' Density N_z =
',num2str(M/(2*zmax))]);
 orient landscape;rotate3d;

figure(d+2)
surf(NRHO2G,ZEEG,RErMagHNT)
title([' Relative Error in Exact to Num Null Test Soln for Mag(H_{\phi}
({\rho} 2)); {\rho} 1 = ',num2str(rho1),' (m) '])
xlabel([' {\rho} 2/{\rho} 1 ; N_{\rho}= ',num2str(Np)]);
ylabel([' z (m) ; N_z = ',num2str(M),' Density N_z =
',num2str(M/(2*zmax))]);
 orient landscape;rotate3d;

figure(d+5)
surf(NRHO2G,ZEEG,RErMagE)
title([' Relative Error in Exact to Num Soln for Mag(E_z ({\rho} 2));
{\rho} 1 = ',num2str(rho1),' (m) '])
xlabel([' {\rho} 2/{\rho} 1 ; N_{\rho}= ',num2str(Np)]);
```

```
ylabel([' z (m) ; N_z = ',num2str(M),' Density N_z =
',num2str(M/(2*zmax))]);
 orient landscape; rotate3d;

figure(d+6)
surf(NRHO2G,ZEEG,RErMagENT)
title([' Relative Error in Exact to Num Null Test Soln for Mag(E_z
({\rho} 2)); {\rho} 1 = ',num2str(rho1),' (m) '])
xlabel([' {\rho} 2/{\rho} 1 ; N_{\rho}= ',num2str(Np)]);
ylabel([' z (m) ; N_z = ',num2str(M),' Density N_z =
',num2str(M/(2*zmax))]);
 orient landscape; rotate3d;

figure(d+7)
plot(DRHO,LSErMagH,'r',DRHO,LSErMagE,'--
g',DRHO,LSErMagHNT,':y',DRHO,LSErMagENT,'-.y')
title([' Least Squares Error for Mag(H_{\phi}) and Mag(E_z) ; {\rho} 1 =
',num2str(rho1),' m '])
xlabel([' {\rho} 2 ; N_{\rho}= ',num2str(Np),' Density N_z =
',num2str(M/(2*zmax))]);
ylabel(' Least Squares Error (%); MagH (solid), MagE (dash), MagH
NullTest (dots), MagE NullTest (dash-dot) ');
grid; orient landscape;

end
```

```
function[EzJ, EzM, HpJ, HpM, Tezj, Tezm, Thpj, Thpm, zOffset] =
dREHJMA(Np,k,rho1,rho2,z1,z2,Jz,Mp)
% Method A for Numerically Calculating the Fields at Rho2.
% Calculate the EzJ, EzM, HpJ, and HpM fields on a cylindrical surface
of
% radius 2 due to the equivilent Electric and Magnetic Currents
% on a cylindrical surface of radius 1. (Radius 2 > Radius 1).
% Fields are calculated by integrating a constant magnitude current
% around a thin ring of radius 1 and summing all the contributing
% pieces of fields from each element in z.
% The Transforms are an interim step to finding the fields.
% Includes ability to calculate fields on a z2 smaller than z1.
% Adapted from M.A. Morgan notes 10 Dec 98

% Calls intphi3 and phisquare2

% Calculate the Phi Integrations for Nphi, zd, Rho1, and Rho2
zd = z1-z1(1);
[u2,u3,u4,u5,v2,v3,v4,v5,dv3,dv4,dv5,w3,w4,w5] =
intphi3(Np,k,rho1,rho2,zd);

% Set Up Integration Matrices from Calculated Nphi segmented phi
% integrations around a thin ring of radius rho1
zn1=length(z1);
zn2=length(z2);
izmax = find(z1>(max(z2)));
zOffset=length(z1(izmax));

U2= phisquare2(u2,zn1,zn2,zOffset);
U3= phisquare2(u3,zn1,zn2,zOffset);
U4= phisquare2(u4,zn1,zn2,zOffset);
U5= phisquare2(u5,zn1,zn2,zOffset);
V2= phisquare2(v2,zn1,zn2,zOffset);
V3= phisquare2(v3,zn1,zn2,zOffset);
V4= phisquare2(v4,zn1,zn2,zOffset);
V5= phisquare2(v5,zn1,zn2,zOffset);
DV3= phisquare2(dv3,zn1,zn2,zOffset);
DV4= phisquare2(dv4,zn1,zn2,zOffset);
DV5= phisquare2(dv5,zn1,zn2,zOffset);
W3= phisquare2(w3,zn1,zn2,zOffset);
W4= phisquare2(w4,zn1,zn2,zOffset);
W5= phisquare2(w5,zn1,zn2,zOffset);


% Calculate the Components due to the Different Current Components
% Method A for solving the fields

dz = abs(z1(2)-z1(1));

% Hp due to Jz (Hp at rho2 due to Hp at rho1)
hpj = (rho2*U3 - rho1*V3) + j*k*(rho2*U2 - rho1*V2);
Thpj = (rho1*dz/(4*pi))*hpj;
HpJ = Jz*Thpj;
clear hpj;

% Ez due to Jz (Ez at rho2 due to Hp at rho1)
```

```
Rezj = ((2+k*k*rho2*rho2)*U3 - 3*rho2*rho2*U5 -
((rho1/rho2)+2*k*k*rho1*rho2)*V3 ...
   + 6*rho1*rho2*V5 + k*k*rho1*rho1*W3 - 3*rho1*rho1*W5);
Imezj = j*k*(2*U2 - 3*rho2*rho2*U4 -(rho1/rho2)*V2 + 6*rho1*rho2*V4 -
3*rho1*rho1*W4);
Tezj = (rho1*dz*30/(j*k))*(Rezj + Imezj);
EzJ = Jz*Tezj;
clear Rezj Imezj;

% Hp due to Mp (Hp at rho2 due to Ez at rho1)
Rhpm = (3*rho1*rho2*U5 - k*k*rho1*rho2*U3
+(2+k*k*rho1*rho1+k*k*rho2*rho2)*V3 ...
   + k*k*DV3 - 3*(rho1*rho1+rho2*rho2)*V5 -3*DV5 -k*k*rho1*rho2*W3 +
3*rho1*rho2*W5);
Imhpm = j*k*(3*rho1*rho2*U4 + 2*V2 - 3*(rho1*rho1+rho2*rho2)*V4 - 3*DV4
+ 3*rho1*rho2*W4);
Thpm = (rho1*dz/(j*k*4*120*pi*pi))*(Rhpm+Imhpm);
HpM = Mp*Thpm;
clear Rhpm Imhpm;

% Ez due to Mp (Ez at rho2 due to Ez at rho1)
ezm = (rho2*V3 - rho1*U3) + j*k*(rho2*V2 - rho1*U2);
Tezm = (rho1*dz/(4*pi))*ezm;
EzM = Mp*Tezm;
clear ezm;
```

```
function [U1,U2,U3,U4,U5,DU3,DU4,DU5,V1,V2,V3,W3,W4,W5] =
intphi4(N,k,rho1,rho2,z2)
% Program uses trapazoidal rule to evaluate integrals on phi=0 to 2pi
circular contour
% of radius rho1 in the x-y plane with field point at (rho2,z2,phi=0) of
even functions:
%     un(phi)= (1/R^n) exp(-jkR)
%     vn(phi)= cos(phi) un(phi)
%     Dun(phi)= (z2^2) un(phi)
% and wn(phi)= cos^2(phi) un(phi)
% where R=sqrt(rho1^2+rho2^2-2rho1*rho2*cos(phi)+z2^2)
% Adapted from IntPhi.m and IntPhi2.m by M.A. Morgan  12/28/98

dp=pi/(N-1); phi=0:dp:pi;  % double the integral from 0 to pi
a=rho1*rho1+rho2*rho2+z2.^2; b=2*rho1*rho2; cp=cos(phi);
A=a.'*ones(1,length(cp)); D=(z2.^2).'*ones(1,length(cp));
CP=ones(length(z2),1)*cp;
R=sqrt(A-b*CP);
u1=exp(-j*k*R)./(R);
u2= u1./R; u3= u2./R; u4= u3./R; u5= u4./R;
v1=CP.*u1; v2=CP.*u2; v3=CP.*u3; v4=CP.*u4; v5=CP.*u5;
w3=CP.*v3; w4=CP.*v4; w5=CP.*v5;
du3=D.*u3; du4=D.*u4; du5=D.*u5;
c=2*dp; ds=c*ones(N,1); ds(1)=c/2; ds(N)=c/2;
U1=u1*ds; U2=u2*ds; U3=u3*ds; U4=u4*ds; U5=u5*ds;
V1=v1*ds; V2=v2*ds; V3=v3*ds;
W3=w3*ds; W4=w4*ds; W5=w5*ds;
DU3=du3*ds; DU4=du4*ds; DU5=du5*ds;
clear A D CP R u* v* w* dv* ds cp phi;
```

```
function A = phisquare4(Vecn,Nz1,Nz2,MC,N,zOffsetMax,zOffsetMin)
%
% Mod4: (length of Intdz1)*N=(length of dz2): Also, use MATLAB
% 'toeplitz' function instead of spdiags.
%
% Mod3: Takes different dz1 | dz2 sizes AND
%              different z1Max | z2Max values
%              into account
%
% Phisquare3 takes a vector un and makes a rectangular diagonal matrix
% out of its elements. 'Vecn(1)' is the main diagonal, 'Vecn(2)'
% is the first diagonal off the main, both above and below
% the diagonal. 'Vecn' is a column vector. After making a
% large rectangular matrix, specific rows are choosen to account
% for the difference in dz1 compared to dz2.
% If they are the dz's are equal, this program makes a banded
% Toeplitz matrix.
% by D.G. Steenman 12/30/98


% Set up elements for each row

U1 = Vecn.';
ANz = length(Vecn);
U = toeplitz(U1,U1);

% First, Deal with Offset
% Second, Choose Selected Columns that correspond to zR2 points
% Third, reshape and add N*MC rows to average the integrations
% Fourth, squeeze and transpose to correct orientation for A

if (zOffsetMax<=0),
   U=U(:,(-zOffsetMax+1):(ANz+zOffsetMin));
else,
   U=U((zOffsetMax+1):(ANz-zOffsetMin),:);
end

U=U(:,((0:Nz2-1)*N + 1));
U=U.';
U=reshape(U,Nz2,N*MC,Nz1);
U=sum(U,2);
U=squeeze(U);
A=U.';
clear U* ANz;
```

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center 8725 John J. Kingman Rd., STE 0944 Ft. Belvoir, VA 22060-6218	2
2.	Dudley Knox Library Naval Postgraduate School 411 Dyer Rd. Monterey, CA 93943-5101	2
3.	Chairman, Code EC Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943-5121	1
4.	Prof. Michael A. Morgan, Code EC/Mw Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943-5121	4
5.	Prof. David C. Jenn , Code EC/Jn Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943-5121	1
6.	Mr. Robert Vitali, Code EC Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943-5121	1
7.	LT Daryl G. Steenman c/o Mr. Larry Steenman 6018 Grand Point San Antonio, TX 78239	3
8.	Dr. Ronald Radlinski Ship Structures & Systems S&T Division, Code 334 Office of Naval Research, Balston Centre Tower One 800 North Quincy Street Arlington, VA 22217-5660	1