AD-A054 007 ARMY ENGINEER TOPOGRAPHIC LABS FORT BELVOIR VA F/G 8/2
INVESTIGATION OF THE APPLICATION OF 'ARRAY ALGEBRA' TO TERRAIN --ETC(U)
APR 78 J R JANCAITIS, R L MAGEE

UNCLASSIFIED ETL-0141 NL

| OF |
AD
A054007

END
DATE
FILMED
6-78
DDC

ETL - 0141

INVESTIGATION OF THE APPLICATION OF "ARRAY ALGEBRA" TO TERRAIN MODELING

James R. Jancaitis   Ronald L. Magee

APRIL 1978

approved for public release; distribution unlimited

U.S. ARMY ENGINEER
TOPOGRAPHIC LABORATORIES
FORT BELVOIR, VA 22060

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER ETL - 0141	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) INVESTIGATION OF THE APPLICATION OF ARRAY ALGEBRA TO TERRAIN MODELING		5. TYPE OF REPORT & PERIOD COVERED Technical Rep[redacted]t
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) James R. Jancaitis Ronald L. Magee		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS U.S. Army Engineer Topographic Laboratories Fort Belvoir, Virginia 22060		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS U.S. Army Engineer Topographic Laboratories Fort Belvoir, Virginia 22060		12. REPORT DATE Apr[redacted]78 13. NUMBER OF PAGES 58
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report investigates the application of array algebra to ETL's terrain modeling procedure in the following manner: 1. analyze array algebra to verify specifically the equivalence of array algebra and the conventional least-squares solutions; 2. analytically and empirically compare the computational efficiency of ETL's terrain modeling algorithm using the current least-squares method and the array algebra technique; and 3. investigate → next page

DD FORM 1473 1 JAN 73 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

403 192

20. continued

the applicability of Rauhala's array algebra to the ETL terrain modeling algorithm. The results showed that the array algebra algorithm is computationally equivalent to the least squares algorithm but has higher implementational overhead. The array algebra algorithm is also less efficient for the ETL terrain modeling problem.

ACCESSION for

NTIS White Section ☒

DDC Buff Section ☐

UNANNOUNCED ☐

JUSTIFICATION..............................

....

BY..

DISTRIBUTION/AVAILABILITY CODES

Dist.	AVAIL. and/or SPECIAL	
A		

PREFACE

This work was supported by the Defense Mapping Agency, Building 56, US Naval Observatory, Washington, DC, under a continuing work effort entitled Investigation of Digital Data Manipulation Techniques. Mr. Bob Penny is the cognizant DMA contact. The constructive criticism provided by Dr. Urho Rauhala and Dr. Duane Brown of DBA, Inc., of Melbourne, Florida, on early drafts of this work is gratefully acknowledged.

CONTENTS

INVESTIGATION OF THE APPLICATION OF "ARRAY ALGEBRA" TO TERRAIN MODELING

by

James R. Jancaitis and Ronald L. Magee

Mapping Developments Division
Topographic Developments Laboratory
US Army Engineer Topographic Laboratories
Fort Belvoir, Virginia 22060

Background.

One of the missions of the Mapping Developments Division of the US Army Engineer Topographic Laboratories (ETL) is to investigate new techniques and equipment which may improve the Defense Mapping Agency's cartographic capabilities. This includes the development of new cartographic products, the improvement of product quality, increasing production rates, and the addition of new capabilities to existing products. One such product now under development at ETL is an improved automated digital terrain modeling procedure based upon the Weighting Function Interpolation Technique (WIT)[1], and which involves the generation of numerous polynomials calculated subject to the least-squares criteria. WIT is an integral part of the software entitled Contouring Via the Surface Averaging Concept, or CONSAC[2], which is ETL's digital terrain modeling and automated contouring computer program. The terrain model constructed by CONSAC includes a rigorous mathematical smoothing of observation errors and results in a compact and continuous functional model over arbitrarily large areas. The contours are produced by analytical solution of the intersection of constant horizontal planes with the functional terrain model which guarantees smooth, continuous, and appropriately semi-parallel lines. The key to the efficiency and applicability of this software is the sequential application of locally valid, simple functional forms. The sequential nature of these algorithms results in the capability to consistently model arbitrarily large regions. The use of simple functional forms for the basic terrain model allows for the efficient derivation of desired terrain information, such as the contour lines. The DMA production applications impacted by this software include:

a. Orthophotographs

b. Automated contouring

c. Weapons systems

d. Databases

This software accomplishes the basic smoothing and modeling tasks by the utilization of least squares linear approximation techniques on the gridded elevation data produced by automatic compilation equipment.[3]

Introduction.

In 1972 Dr. Rauhala[4] introduced an improved algorithm for linear approximation of data lying on an orthonormal grid subject to the least-squares criteria and subsequently included it in his 1976 "Array Algebra" manuscript.[5] This work and the further investigations done by Kratky (1976)[6] and Crombie (1976 a and b)[7,8] came to our attention in June 1976.

Kratky stated that his efforts involving polynomial transformation of satellite imagery enjoyed a significant increase in computational efficiency and accuracy as a result of the inclusion of this new modification to the least-squares algorithm. Based upon the pioneering work of Rauhala and the subsequent success of Kratky and Crombie, this technique was considered to possibly have a cost-effective application to the approximation algorithm in ETL's terrain modeling procedure.

A direct implementation was not undertaken immediately due to some uncertainties we had in the theoretical basis and applicability of this technique to our problem. At that time, Rauhala's proof of the array algebra algorithm's equivalence to the conventional least-squares algorithm was difficult to follow due in part to poor English language translations of Rauhala's work, and in part to his deviation from the simpler currently accepted matrix notational forms in those cases where they could have been utilized. At ETL the proof of these algorithms' equivalence was first investigated by Crombie (1976 a), who produced a lengthy derivation of their algebraic equivalence using a laborious presentation and manipulation of the basic system of scalar equations. Of concern to us was the fact that the derivation of the array algebra approximation algorithm did not proceed from an a priori stated mathematical condition, as does the least-squares normal equations, splines, and all other well-used approximation algorithms. Further, the computational efficiency described by Kratky, Rauhala, and Crombie (1976 b) involved cases similar to, but not exactly like, our application.

For the above reasons, it was determined that the application of array algebra to ETL's terrain modeling procedure was to be investigated in the following manner:

1. A theoretical analysis of array algebra would be performed in order to verify specifically the equivalence of array algebra and the conventional least-squares solutions for our application.

2. A paper study would be conducted to compare the computational efficiency of ETL's terrain modeling algorithm using the current conventional least-squares method and the array algebra technique.

3. An empirical study involving the development of two digital computer programs would be conducted to verify the paper study results, and to evaluate those aspects of computer implementation not easily predicted, such as FORTRAN "DO-LOOP" overhead.

4. Given that array algebra proved to be a cost-effective technique for ETL's terrain modeling procedure, it would then be implemented into CONSAC.

In the following sections, this paper reports the status and results of these investigations. In order to facilitate the presentation of this paper, all of the mathematical derivations have been placed in the appendixes with only the major results contained in the body.

Theoretical Analysis[9]

As stated in the introduction, one of our major initial concerns regarding the "array algebra" algorithm was its mode of derivation.

In contrast to all other widely used numerical algorithms[10], the array algebra solution did not proceed from an a priori stated mathematical condition. Rauhala's derivation of the solution equations for the simplest, two-dimensional case, involves a utilization of the bilinear form in concert with only matrix manipulations. (see appendix C). This method of derivation is unilluminating in terms of what this solution does for the problem solver employing it. The previously mentioned proofs of equivalence of the array algebra and least-squares solutions are equally unilluminating derivations. For these reasons, this effort was directed toward a proof and verification of these solutions equivalence which followed more traditional lines. More specifically, a derivation of the two-dimensional, array algebra solution was sought, which was a systematic derivation of the numerical solution that is easily seen and proceeds from an a priori stated mathematical condition using direct and simple analyses based upon widely known and understood basic principles.

Such a derivation was found and will now be presented in the following format. On the left-hand side of the page the well-known derivation of the least-square's normal equations[11] will be presented, and on the right-hand side, the corresponding steps of our new derivation of the two-dimensional array algebra solution of interest in our application. (For a complete definition of all matrices see appendixes A and C). First, a model equation is chosen, with the array algebra's bilinear form dependent upon the observations lying on an orthonormal grid in some coordinate system,

$$Z_L = AC_L \qquad\qquad W = XC_AY^T$$

and the objective function formed

$$\phi = (Z_L - AC_L)^T(Z - AC_L) \qquad\qquad \phi = \| W - XC_AY^T \|$$

then minimized by differentiation

$$\frac{\partial\phi}{\partial C_L} = (Z_L - AC_L)^T(-2A) \qquad\qquad \frac{\partial\phi}{\partial C_A} = X^T(W - XC_AY^T)Y(-2)$$

setting these partials equal to zero and solving by rearranging and assuming the existence of inverses gives

$$C_L = (A^TA)^{-1}A^TZ_L \qquad\qquad C_A = (X^TX)^{-1}X^TWY(Y^TY)^{-1}$$

The one-to-one correspondence of steps and similarity in dependence upon the application of calculus of variations results in a much more illuminating theoretical basis for the array algebra solution. This new derivation also represents a much clearer proof of the equivalence of these algorithms because a simple expansion of the model and objective functions to their basic scalar equation form demonstrates their equality. As is well-known, two solutions of an approximation problem are equivalent provided the model equations and objective functions are equal. The derivation presented here also allows for an easy and straightforward inclusion of "structured" observation weighting and arbitrary control constraints. The modification of the algorithm for Kalman Filter[12] processing has not yet been made but seems to be a definite and useful possible extension at this time. Appendix A contains a complete and detailed presentation of our derivation of the two-dimensional array algebra solution, as well as the derivation when "structured" observational weighting and observation constraints are included. Appendix B contains the detailed derivation of the standard least-squares normal equations, as well as a numerical example. Appendix C contains a detailed presentation of Rauhala's derivation of the two-dimensional array algebra solution, as well as the same numerical example of appendix B, only this time solved by the array algebra algorithm.

In conclusion, the two-dimensional array algebra solution of Rauhala represents another computational procedure for approximation subject to the least-squares critera when the observations lie on an orthonormal grid in some coordinate system. We have presented a derivation of this algorithm which follows more traditional lines and is more illuminating with regard to what, exactly, the array algebra algorithm does for the problem solver utilizing it.

Computational Requirements Analysis[13]

The computational requirements analysis was completed in October 1976, and provided some insight as to the impact array algebra would have on ETL's terrain modeling algorithm, CONSAC II. In order to isolate the computational differences between the conventional least-squares and array algebra techniques we essentially performed a quantitative comparison of the number of computer multiplications and additions needed to compute the coefficient matrices C_L or C_A where:

$$C_L = (A^TA)^{-1}A^T Z_L, \text{ the least-squares normal equation}$$

or

$$C_A = (X^TX)^{-1}X^T Z_A \; Y(Y^TY)^{-1} \quad , \text{ the array algebra solution}$$

and: Z_L and Z_A are matrices of observation

A is a matrix of polynomial terms

X and Y are uni-directional parameter matrices

(for further specification see appendixes B and C).

Since each C_L or C_A describes only one polynomial, then it would seem that it must be calculated once for every polynomial needed to describe an entire map sheet. Fortunately, this is not true. Since $(A^TA)^{-1}A^T$ is dependent only upon the x-y coordinates of each of the observations being fit, then $(A^TA)^{-1}A^T$ must be calculated whenever the x-y location of the set of observations change. If, however, we choose to move the origin of the entire coordinate system commensurately with our progression through the set of observations, then $(A^TA)^{-1}A^T$ is constant and independent of the position of the origin. Thus, use of this local roving coordinate system allows us to calculate $(A^TA)^{-1}A^T$

only once per map sheet. The same technique prevails for $(X^TX)^{-1}X^T$ and $Y(Y^TY)^{-1}$.

Now it can be seen that both array algebra and the conventional least-squares technique require two computational events: one-time operations and repetitive operations. (see figure 1.)

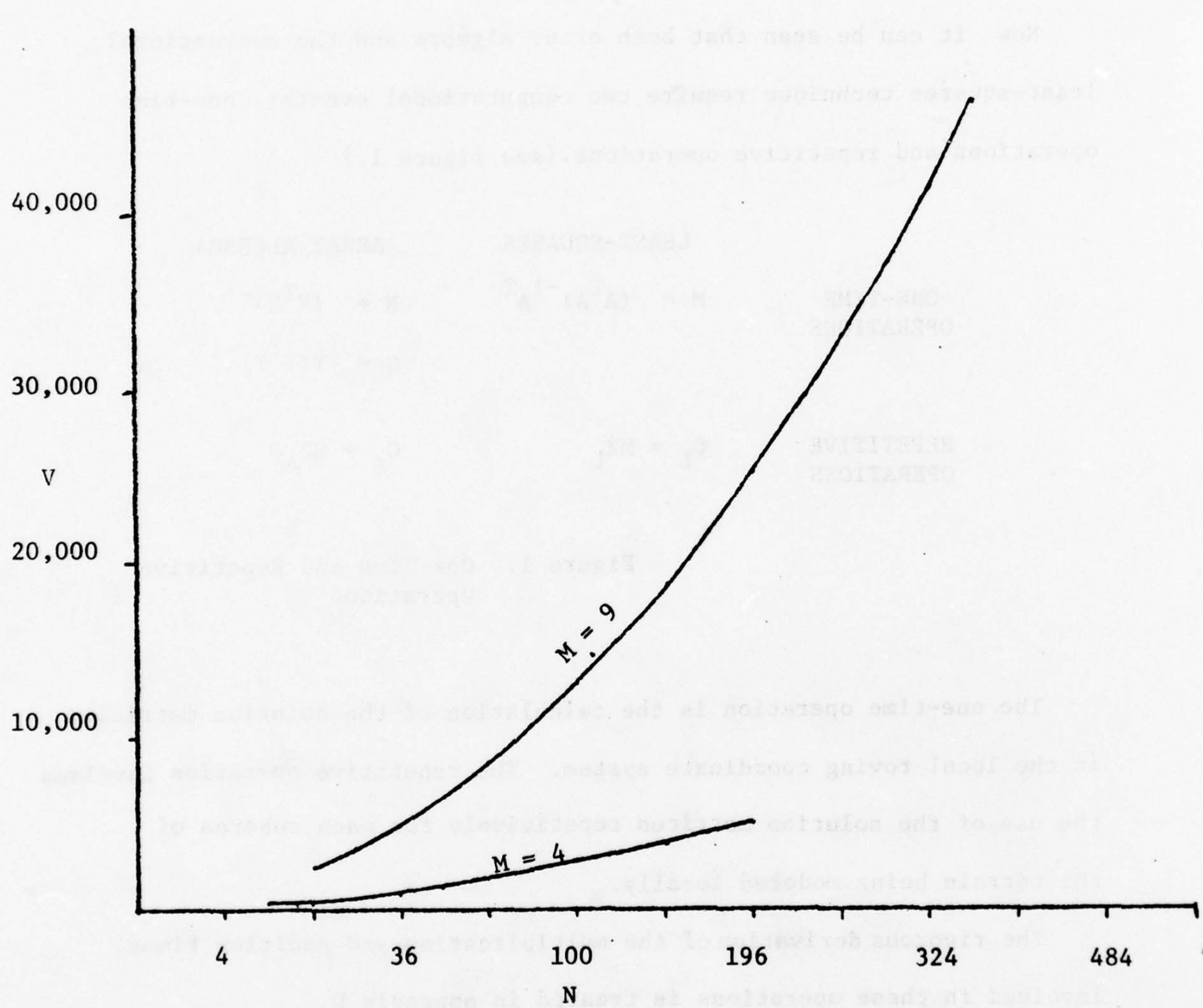
	LEAST-SQUARES	ARRAY ALGEBRA
ONE-TIME OPERATIONS	$M = (A^TA)^{-1}A^T$	$N = (X^TX)^{-1}X^T$ $Q = Y(Y^TY)^{-1}$
REPETITIVE OPERATIONS	$C_L = MZ_L$	$C_A = NZ_AQ$

Figure 1. One-Time and Repetitive Operations

The one-time operation is the calculation of the solution matricies in the local roving coordinate system. The repetitive operation involves the use of the solution matrices repetitively for each subarea of the terrain being modeled locally.

The rigorous derivation of the multiplication and addition times involved in these operations is treated in appendix D.

A calculation of the difference of the number of one-time multiplications required by the conventional least-squares method and the array-algebra method produced the results shown in figure 2a. Notice that the computational savings increase with the size of the model polynomial and with the number of elevation observations per polynomial fit. One-time



V = The difference of least-squares one-time multiplications and array algebra one-time multiplications.

N = Number of observations per polynomial

M = Number of terms in model polynomial

Figure 2a: Difference of one-time multiplications addition is similar and is shown as figure 2b.

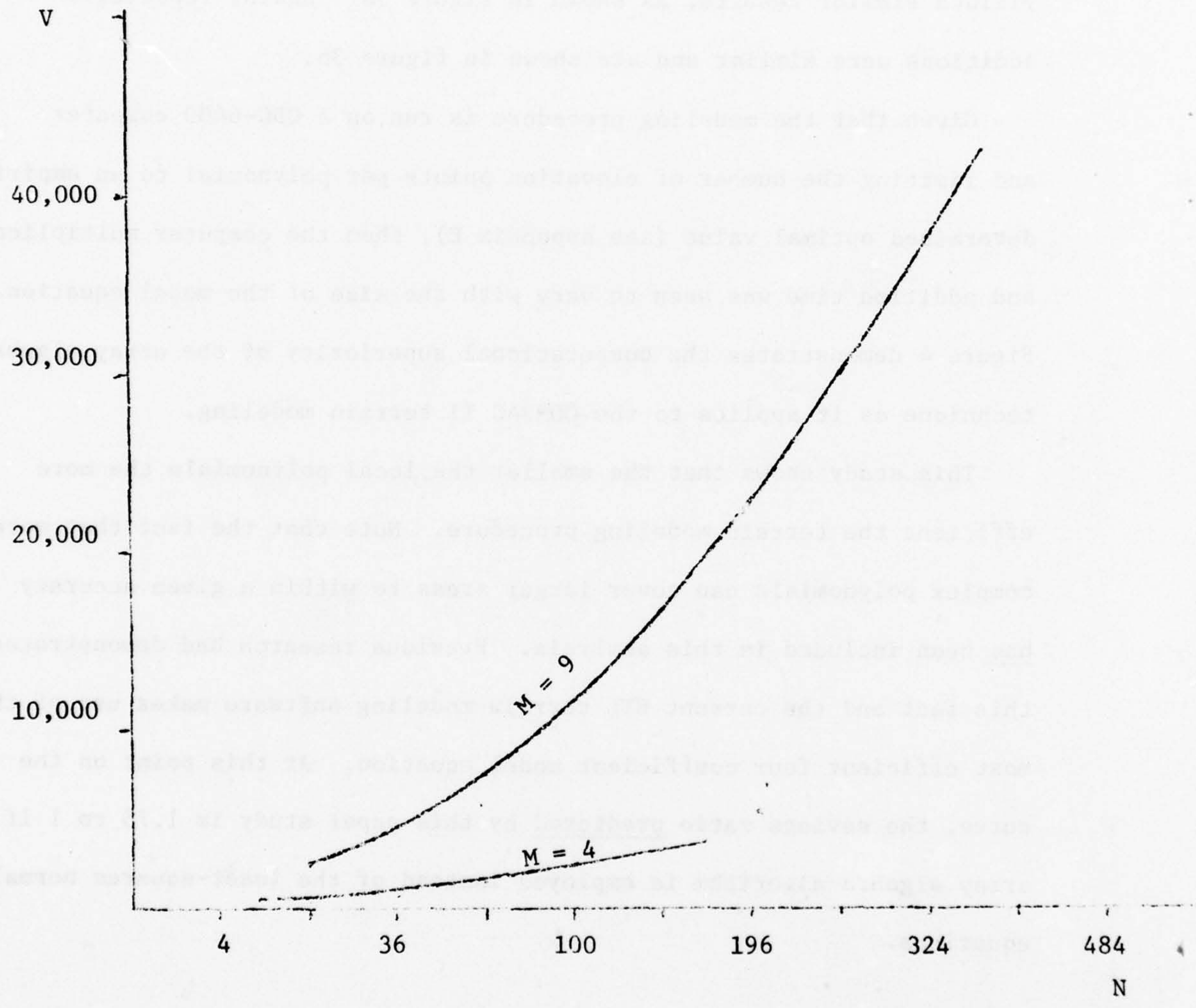
The same difference concerning total repetitive multiplications yielded similar results, as shown in figure 3a. Again, repetitive additions were similar and are shown in figure 3b.

Given that the modeling procedure is run on a CDC-6400 computer and limiting the number of elevation points per polynomial to an empirically determined optimal value (see appendix E), then the computer multiplication and addition time was seen to vary with the size of the model equation. Figure 4 demonstrates the computational superiority of the array algebra technique as it applies to the CONSAC II terrain modeling.

This study shows that the smaller the local polynomials the more efficient the terrain modeling procedure. Note that the fact that more complex polynomials can cover larger areas to within a given accuracy has been included in this analysis. Previous research had demonstrated this fact and the current ETL terrain modeling software makes use of the most efficient four coefficient model equation. At this point on the curve, the savings ratio predicted by this paper study is 1.75 to 1 if the array algebra algorithm is employed instead of the least-squares normal equations.

Empirical Analysis.

In order to more thoroughly investigate these algorithms' relative computation requirements an empirical study was undertaken. This empirical study was deemed necessary to fully and unambiguously compare



V = The difference of least-squares one-time additions and array algebra one-time additions.

N = Number of observations per polynomial

M = Number of terms in model polynomial

Figure 2b: Difference of one-time additions

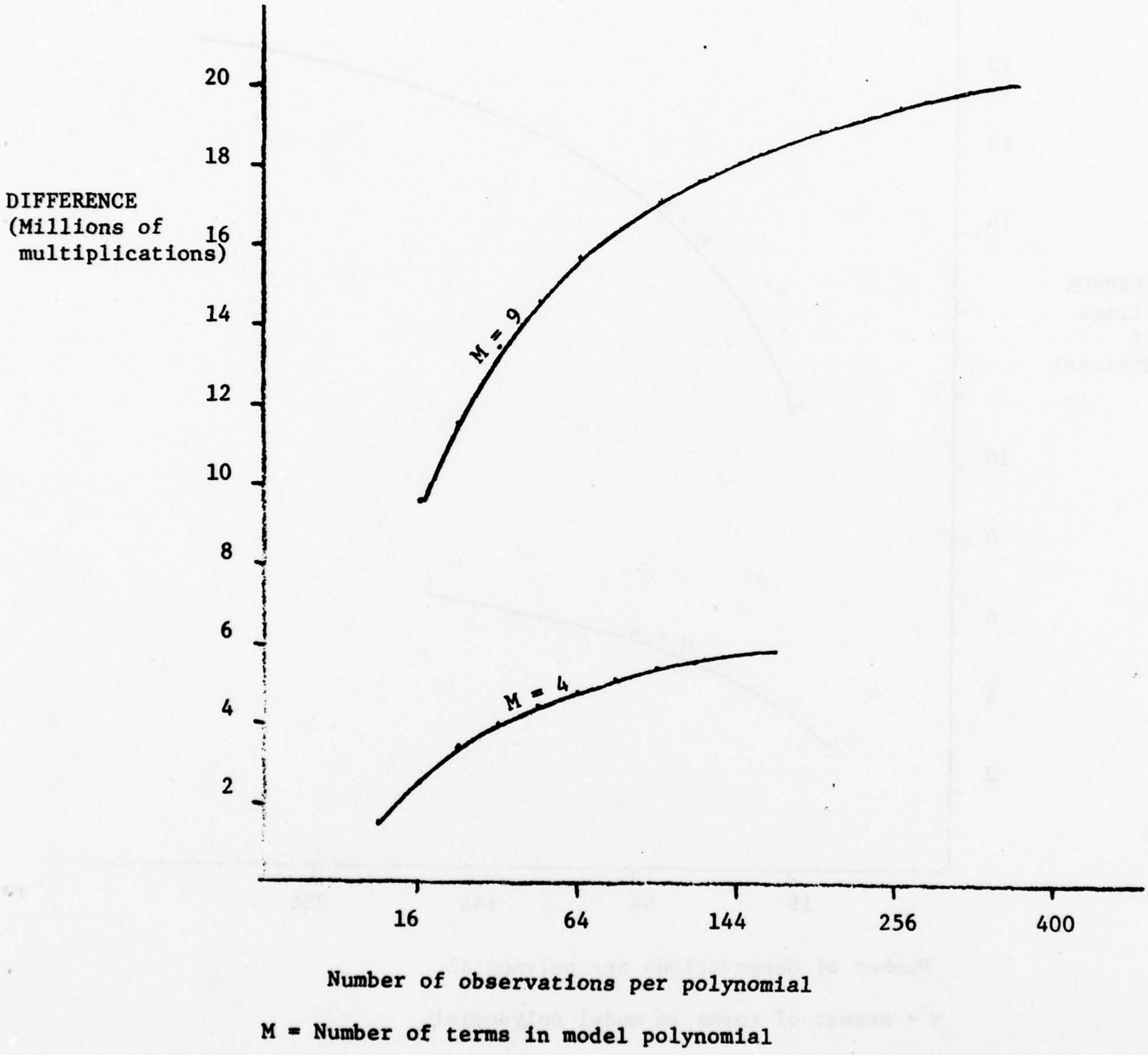


Figure 3a: Difference of Least-Squares and Array Algebra repetitive multiplications per map sheet

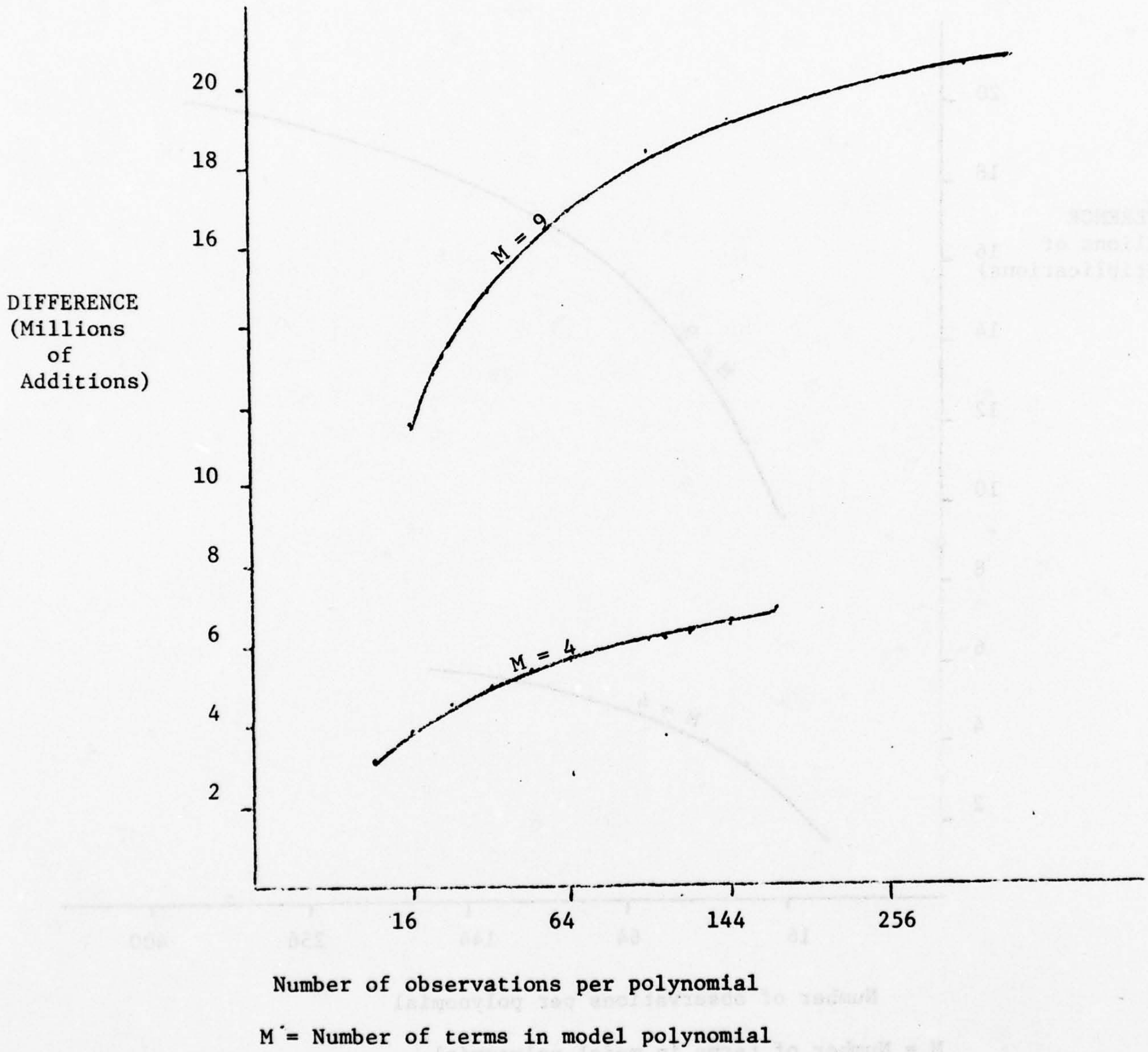


Figure 3b: Difference of Least-Squares and Array Algebra repetitive additions per map sheet

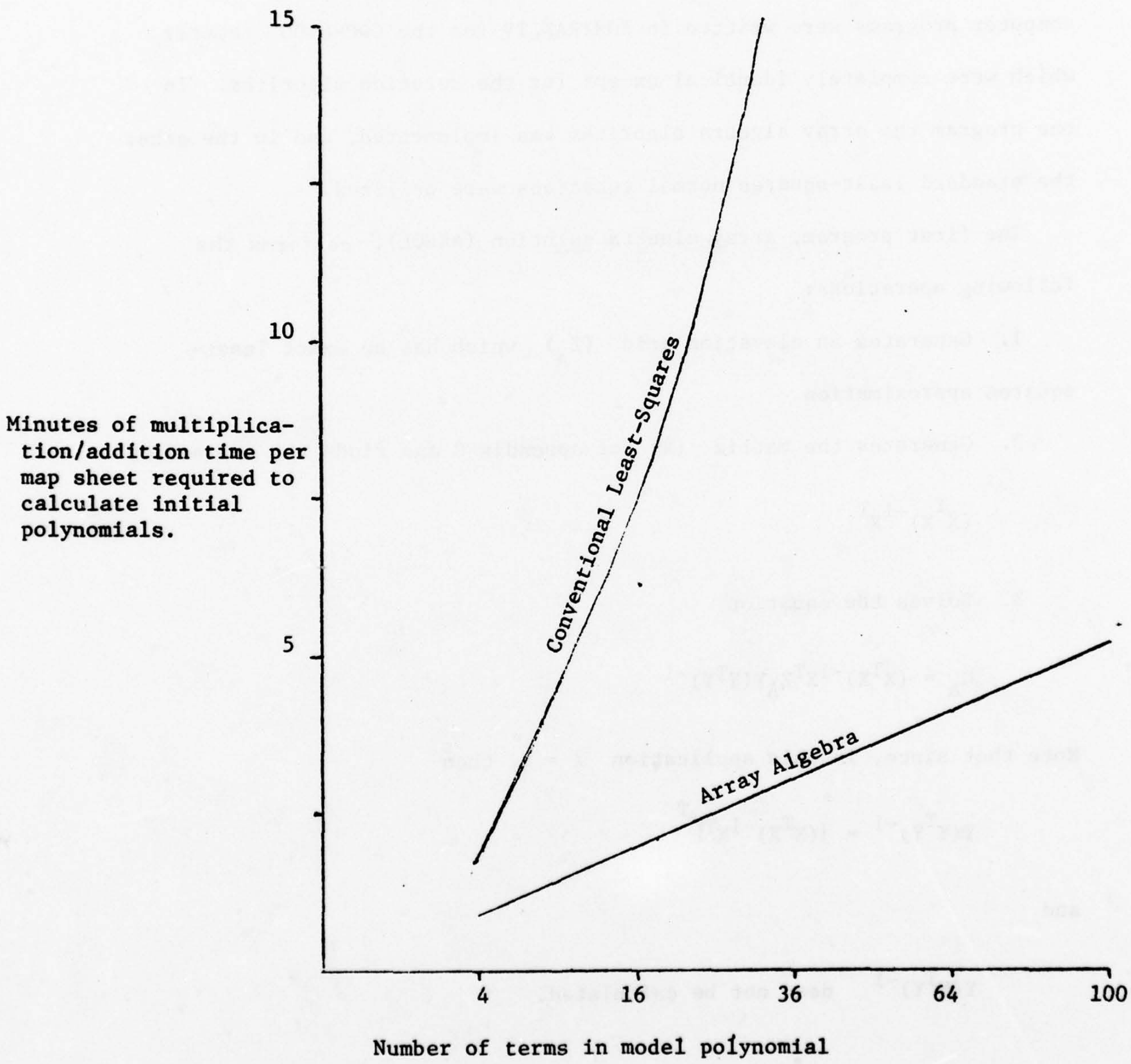


Figure 4: Theoretical comparison of efficiency: Least-Squares vs Array Algebra

those aspects of these algorithms which are not as easily studied analytically. The type of factors referred to include FORTRAN do-loop overhead, computer data access times, etc. To this end, two digital computer programs were written in FORTRAN IV for the CDC-6400 computer which were completely identical except for the solution algorithm. In one program the array algebra algorithm was implemented, and in the other the standard least-squares normal equations were utilized.

The first program, array algebra solution (AASOL), performs the following operations:

1. Generates an elevation grid (Z_A) which has an exact least-squares approximation.

2. Generates the matrix (X) of appendix C and finds the expression

$$(X^T X)^{-1} X^T$$

3. Solves the equation

$$C_A = (X^T X)^{-1} X^T Z_A Y (Y^T Y)^{-1}$$

Note that since, in this application $X = Y$, then

$$Y(Y^T Y)^{-1} = [(X^T X)^{-1} X^T]^T$$

and

$Y(Y^T Y)^{-1}$ need not be calculated.

4. Evaluates the C_A matrix of coefficients and produces a set of derived elevation data. This data is then compared to Z_A and a sum-squared error computed.

5. Measures the CP times of the one-time run of step (2) and the repetitive run of step (3), and displays the total computation time needed to model a complete 1:50,000 map sheet.

Successful runs were achieved for model equations of 4, 9, 16, 25, 36, 49, 64, 81, and 100 terms, with computation times exceeding the multiplication and addition times (as predicted in the computational analysis) by a factor of approximately 2.7. The actual CP execution times ran from 166 seconds for the 4-term model polynomial to 806 seconds for the 100-term model.

The conventional Least-Squares Solution test routine, LSSOL, performed functions equivalent to the Array Algebra Solution test routine. Successful runs, however, could be achieved only for model polynomials of 4, 9, and 16 terms. Any attempt to dimension the routine to accept a larger model equation exceeded the storage capabity of the CDC-6400. Computation times exceeded the predicted multiplication and addition times by a factor of 2.3. Actual CP execution time ran from 218 seconds for the 4-term model to 941 seconds for the 16-term model. The comparison of the computation times in AASOL and LSSOL are presented in figure 5.

The savings ratio actually achieved for the most efficient 4-coefficient model case of our terrain modeling procedure was 1.3 to 1, more than 50 percent lower than the theoretically projected ratio of

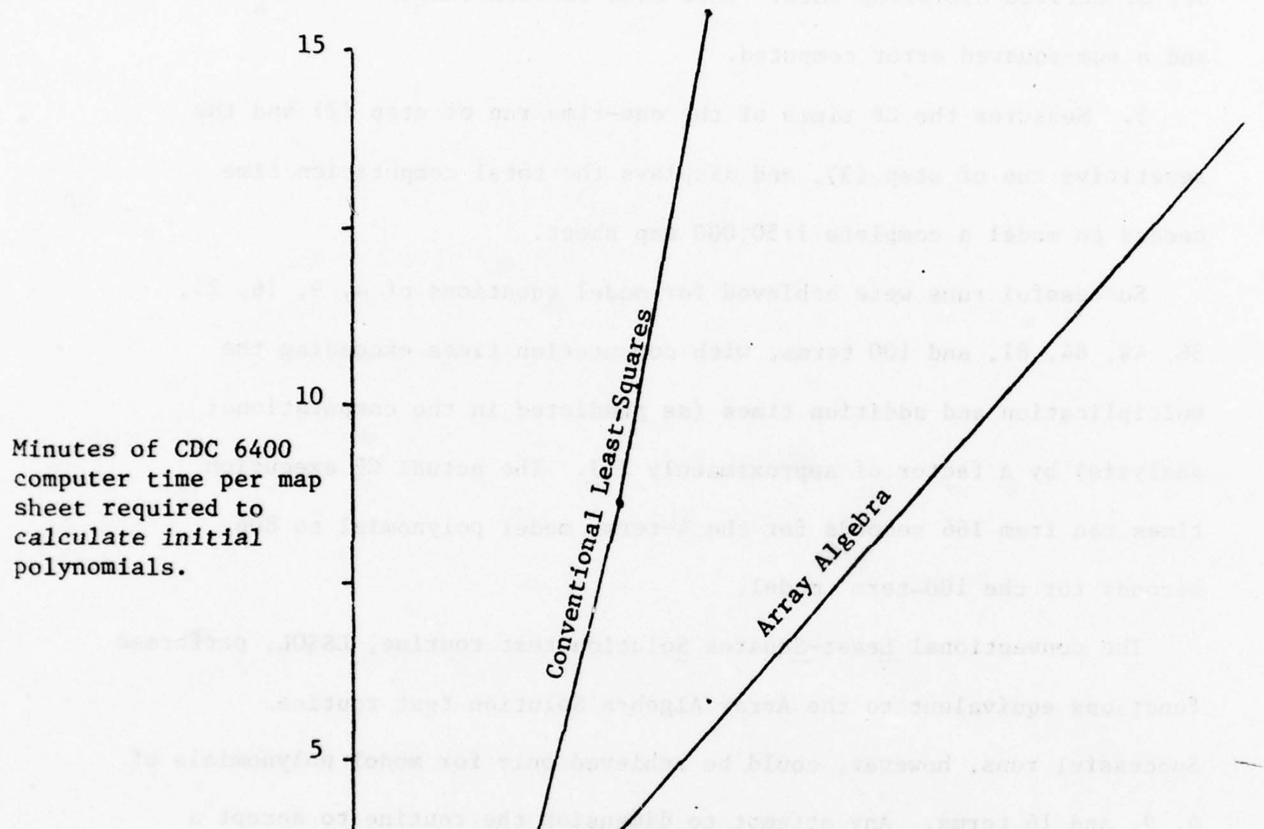


Figure 5: Empirical comparison of efficiency: Least-Squares vs. Array Algebra

1.75 to 1. This difference is caused by the empirically determined higher implementation overhead associated with the array algebra algorithm--2.3 times the least-squares projection vs. 2.7 times the array algebra projection. The higher implementation overhead of array algebra appears to be due to the much larger number of do-loops required. The major result, however, is that the array algebra approximation algorithm is more efficient in all cases.

Examining the possible savings at a production center if the array algebra algorithm were used in the most efficient case:

1. 52 seconds were saved per map sheet
2. 500 map sheets are produced per year[14]
3. $160 per hour is the current cost of computer processor time at a production center. Implementation of the array algebra algorithm will therefore result in an annual savings of $1,155.56. Since our best estimates indicate that array algebra will require at least three man-months to implement in CONSAC II, such an implementation cannot be considered cost-effective when viewed only in terms of its computational efficiency. There is an effort presently under way, however, to modify CONSAC II to utilize a 9-term polynomial model in order to investigate possible improvement of contour fitting and expression. Array algebra can be economically included in this effort and, if used in production, will result in considerably greater savings than if used with a 4-term model.

Conclusions.

An investigation of the applicability of Rauhala's array algebra to the ETL terrain modeling algorithm has been completed. Our theoretical analysis of the two-dimensional bilinear polynomial case has verified its equivalence with the least-squares normal equations for orthonormal observation grids and provides for a more traditional and illuminating derivation. The comparison of the computational requirements paper study and the empirical analysis has shown that:

1. the array algebra algorithm has significantly more implementational overhead than the least-squares algorithm.

2. the array algebra algorithm is more efficient than the least-squares algorithm in all cases pertinent to our requirements.

3. for our currently employed, most efficient terrain modeling case, array algebra has an improvement ratio of 1.3 to 1, amounting to a savings of only about $1,000 per fiscal year in a production environment.

Based on our analyses, implementation of the array algebra algorithm in the specific case investigated is not as significant as had been anticipated.

Our current plans are to proceed with the array algebra implementation at the same time that we are expanding the terrain modeling software's capability under a related R&D effort[15]. Modification of our terrain modeling software to the array algebra algorithm can be economically included in planned modifications under that work effort giving us the opportunity to gain further experience with this powerful algorithm, which we feel can be profitably employed whenever approximation of observations on an orthonormal grid is required.

REFERENCES

1. Jancaitis, J.R. and Junkins, J.L., "Modeling in N-Dimensions Using a Weighting Function Approach," Journal of Geophysical Research, Vol. 79, No. 23, August 10, 1974, pgs. 3361-3366

2. Jancaitis, J.R., "Modeling and Contouring Irregular Surfaces Subject to Constraints," US Army Engineer Topographic Laboratories, Final Technical Report, No. ETL-CR-74-19, January 1975, 174 pages.

3. Bertram, S., "The Universal Automatic Map Compilation Equipment," presented to the Tenth Congress of the International Society of Photogrammetry, Lisbon, Portugal, September 1964, and Photogrammetric Engineering, March 1975.

4. Rauhala, U. "New Solutions for Fundamental Calibration and Triangulation Problems," Svensk Lantmateritidskrift, Vol 4, No. 2, 1972.

5. Rauhala, U., "A Review of Array Algebra," Fotogrametriska Meddelanden," 2:38, Department of Photogrammetry, Royal Institute of Technology, Stockholm, Sweden, 1976.

6. Kratky, V., "Grid Modified Polynomial Transformation of Satellite Imagery," Remote Sensing of Environment, No. 5, pgs. 67-74, 1976.

7. Crombie, M., "Equivalence of Least-Squares and Array Algebra," ETL Working Paper, 6 pages, July 1976a.

8. Crombie, M., "Array Algebra - Multilinear Forms," ETL Working Paper, 26 pages, June 1976b.

9. Jancaitis, J. "Theoretical Analysis of Array Algebra," Memo for Record, US Army Engineer Topographic Laboratories, Fort Belvoir, VA, October 7, 1976, 11 pages.

10. Henrice, P., Elements of Numerical Analysis, John Wiley and Sons, Inc., New York, 1974, 336 pages.

11. Noble, B., Applied Linear Algebra, Prentice Hall, Inc., Englewood Clifts, New Jersey, 1969, 523 pages.

12. Guria, I.A., "An Algebraic Approach to Optimal State Estimation," Huges Aircraft Company, Space Systems Division, SSD70072R, March 1967, 93 pages.

13. Magee, R.L., "The Use of Array Algebra in Terrain Modeling Procedures," US Army Engineer Topographic Laboratories Technical Report, October 1976, 93 pages.

14. Finley, C.G. "SACARTS Status Study," DMATC Staff Study, September 9, 1975, 25 pages.

15. Jancaitis, J.R., "Elevation Data Compaction by Polynomial Modeling," presented to the 1977 Annual Spring Joint Meeting of the ASP-ACSM, Washington-Hilton Hotel, Washington, DC, March 1, 1977.

APPENDIX A. THEORETICAL ANALYSIS OF ARRAY ALGEBRA AS APPLIED TO ETL'S TERRAIN MODELING ALGORITHMS

Preliminaries. The problem statement proceeds as follows: It is desired to approximate the q by p set of observations,

$$W_{ij} \text{ at } X_i, Y_j \text{ for } \begin{array}{l} i = 1, 2, 3, \ldots, q \\ j = 1, 2, 3, \ldots, p \end{array}$$

where the X_i and Y_j are the independent variable locations of the measured observations, (It is important to note that the W_{ij} lie on an ortho-normal grid in the plane of the independent variables X,Y by construction!) with the model equation.

$$W_{ij} = \sum_{r=0}^{T} \sum_{s=0}^{U} C_{rs} X_i^r Y_j^s .$$

Where the C_{rs} are the unknown approximating function's coefficients, where (T+1) times (j+1) is less than Q times P. It is now evident that the model equation can be written in matrix form (using the bilinear form) as:

$$\underset{1 \times 1}{W_{ij}} = \underset{1 \times (T+1)}{[1, X_i, X_i^2, \ldots, X_i^T]} \underset{(T+1) \times (U+1)}{\begin{bmatrix} C_{00}, C_{01}, \ldots, C_{0U} \\ C_{10}, C_{11}, \ldots, C_{LU} \\ \cdot \quad \cdot \quad \cdot \\ \cdot \quad \cdot \quad \cdot \\ \cdot \quad \cdot \quad \cdot \\ C_{T0}, C_{T1}; \ldots, C_{TU} \end{bmatrix}} \underset{(U+1) \times 1}{\begin{bmatrix} 1 \\ Y_j \\ Y_j^2 \\ \cdot \\ \cdot \\ \cdot \\ Y_j^U \end{bmatrix}}$$

where the size of the matrices is shown underneath each matrix.

Further, it is possible to define observational matrices which include all of the observations, e.g.,

$$W = \begin{bmatrix} W_{11}, & W_{12}, & \ldots, & W_{1p} \\ W_{21}, & W_{22}, & \ldots, & W_{2p} \\ \vdots & \vdots & & \vdots \\ W_{q1}, & W_{q2}, & \ldots, & W_{qp} \end{bmatrix}$$

and then the entire set of model, observation equations can be written in matrix form as:

$$\underset{q \times p}{W} = \underset{q \times (T+1)}{X} \; \underset{(T+1) \times (U+1)}{C} \; \underset{(U+1) \times p}{Y^T}$$

where

$$X = \begin{bmatrix} 1, & X_1, & X_1^2, & \ldots, & X_1^T \\ 1, & X_2, & X_2^2, & \ldots, & X_2^T \\ \vdots & \vdots & \vdots & & \vdots \\ 1, & X_q, & X_q^2, & \ldots, & X_q^T \end{bmatrix}$$

and

$$Y = \begin{bmatrix} 1, & Y_1, & Y_1^2, & \ldots, & Y_1^U \\ 1, & Y_2, & Y_2^2, & \ldots, & Y_2^U \\ \vdots & \vdots & \vdots & & \vdots \\ 1, & Y_p, & Y_p^2, & \ldots, & Y_p^U \end{bmatrix}$$

One other preliminary definition necessary in the following sections is the differentiation of an arbitrary scalar function, with respect to an arbitrary rectangular matrix, D , where

$$D = \begin{bmatrix} D_{11}, & D_{12}, & D_{13}, & \ldots, & D_{1l} \\ D_{21}, & D_{22}, & D_{23}, & \ldots, & D_{2l} \\ \vdots & \vdots & \vdots & & \vdots \\ D_{m1}, & D_{m2}, & D_{m3}, & \ldots, & D_{ml} \end{bmatrix}$$

as is customary the matrix of partials is denoted as:

$$\frac{\partial\beta}{\partial D} = \begin{bmatrix} \frac{\partial\beta}{\partial D_{11}}, & \frac{\partial\beta}{\partial D_{12}}, & \ldots, & \frac{\partial\beta}{\partial D_{1l}} \\ \frac{\partial\beta}{\partial D_{21}}, & \frac{\partial\beta}{\partial D_{22}}, & \ldots, & \frac{\partial\beta}{\partial D_{2l}} \\ \vdots & \vdots & & \vdots \\ \frac{\partial\beta}{\partial D_{m1}}, & \frac{\partial\beta}{\partial D_{m2}}, & \ldots, & \frac{\partial\beta}{\partial D_{ml}} \end{bmatrix}$$

4. Derivation of Array Algebra Solution: There are an infinite number of solutions to the approximation problem as stated in section three. A unique solution can only be achieved if additional conditions are specified. The condition which has been determined (by the study reported upon herein) to result in Rauhala's *array* algebra solution is as follows:

Determine the coefficients of the model equation, C_{rs}, which minimize the scalar function, ϕ, defined as:

$$\phi = \| W - X\, C\, Y^T \|$$

where the symbol $\| \cdots \|$ *is the squared matrix norm*, which is defined for an arbitrary rectangular mxl matrix, D, as

$$\| D \| = \sum_{i=1}^{m} \sum_{j=1}^{l} D_{ij}^2 \, .$$

The careful reader will note that this is a simple reformulation of the scalar objective function utilized to derive the standard least-squares solution!

This reformulation of the least-squares objective function simply utilized the bilinear form characteristic of and required by array algebra and the well-known concept of matrix norms.

Proceeding with the derivation, the objective function is minimized by differentiating it with respect to the unknown model coefficients, C_{rs} and setting the result equal to zero:

$$\frac{\partial \phi}{\partial C_{ij}} = \frac{\partial}{\partial C_{ij}} \sum_{k=0}^{T} \sum_{l=0}^{U} \theta_{kl}^2$$

where:

$$\theta_{kl} = W_{kl} - [1, X_k, \ldots X_k^T] \; C \begin{bmatrix} 1 \\ Y_l \\ \cdot \\ \cdot \\ \cdot \\ Y_l^U \end{bmatrix}$$

then:

$$\frac{\partial \phi}{\partial C_{ij}} = \sum_{k=0}^{T} \sum_{l=0}^{U} 2\theta_{kl} \frac{\partial \theta_{kl}}{\partial C_{ij}} = \sum_{k=0}^{T} \sum_{l=0}^{U} 2\theta_{kl} (-X_k^i Y_l^j)$$

$$= -2 \sum_{k=0}^{T} \sum_{l=0}^{U} X_k^i \theta_{kl} Y_l^j$$

$$= -2 (X_1^i, X_2^i, \ldots, X_T^i) [W - X \, CY^T] \begin{bmatrix} Y_1^j \\ Y_2^j \\ \cdot \\ \cdot \\ \cdot \\ Y_U^j \end{bmatrix}$$

and

$$\frac{\partial \phi}{\partial C} = -2 \; X^T (W - X \, CY^T) Y$$

and

$$-2\ X^T[W - X\ C\ Y^T]Y = 0$$

multiplying the terms out, dividing by -2 and rearranging gives:

$$X^TX\ C\ Y^TY = X^T\ W\ Y$$

postulating the existence of $(X^TX)^{-1}$ and $(Y^TY)^{-1}$, and multiplying them on the left and right respectively gives:

$$C = (X^TX)^{-1}\ X^T\ W\ Y\ (Y^TY)^{-1}$$

which is exactly Rauhala's array algebra solution! It is now obvious that the array algebra solution must give exactly the same numerical results as the least-squares solution for those approximation problems which have their observations on an orthonormal gird--they must, they are both the unique solution of over-determined systems subject to exactly the same conditions!

5. Approximation with Orthonormal Observational Weighting: The capability for nearly arbitrary weighting of the observations is a requirement for many of the MC&G approximation applications. After considerable analysis, it was determined that completely arbitrary weighting of the observations was impossible if the

bilinear form and array algebra solution were used. A compromise, orthonormal weighting, however, was successfully found which meets the observational weighting requirements of our applications. By orthonormal weighting, it is meant that a single unique weight can be associated with each x and each y coordinate specifying the observation's ortho-normal grid. The weight corresponding to any single observation is the product of its x-weight and y-weight. This scheme is implemented as follows: The objective function is modified to:

$$\phi = \left\| \Omega_x \, (W - X\,C\,Y^T) \, \Omega_y \right\|$$

where

$$\Omega_x = \begin{bmatrix} \omega_{x_1}, & 0, & 0, & \dots, & 0 \\ 0, & \omega_{x_2}, & 0, & \dots, & 0 \\ 0, & 0, & \omega_{x_3}, & \dots, & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0, & 0, & 0, & \dots, & \omega_{x_q} \end{bmatrix}$$

$$\Omega_y = \begin{bmatrix} \omega_{y_1}, & 0, & 0, & \dots, & 0 \\ 0, & \omega_{y_2}, & 0, & \dots, & 0 \\ 0, & 0, & \omega_{y_3}, & \dots, & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0, & 0, & 0, & \dots, & \omega_{y_p} \end{bmatrix}$$

and the weighting associated with the i,j-th observation is given by

$$\omega_{ij} = \omega_{x_i} \omega_{y_j}$$

This scheme, while not completely arbitrary, does allow for great flexibility in weighting, with the observation weighting over the rectangular area being the product of the x and y variances. The derivation for the weighted solution now follows as:

objective function: $\phi = \left\| \Omega_x (W - X C Y^T) \Omega_y \right\|$

minimize by setting partials with respect to unknowns to zero;

$$\frac{\partial \phi}{\partial C} = -2 X^T \Omega_x [W - X C Y^T] \Omega_y Y = 0$$

rearranging:

$$X^T \Omega_x W \Omega_y Y - X^T \Omega_x X C Y^T \Omega_y Y = 0$$

postulating existence of inverses:

$$C = (X^T \Omega_x X)^{-1} X^T \Omega_x W \Omega_y Y (Y^T \Omega_y Y)^{-1}$$

This solution, as in the unweighted case of section four, must produce exactly the same numerical results as the standard least-squares solution with the same weighting--because both techniques produce the unique solution based on exactly the same mathematical critera.

6. Constrained Array Algebra: As mentioned earlier, it is often necessary to rigorously constrain the approximating function's value at more precisely known points. Utilizing the matrix norm/bilinear form objective function and the calculus of variations with Lagrange multipliers, observation constraints can be rigorously incorporated into the array algebra solution.

The derivation of the constrained, weighted array algebra solution proceeds follows:

minimize the objective function; ϕ ,

$$\phi = \left\| \Omega_x \; (W - X\,C\,Y^T)\Omega_y \right\|$$

subject to the N constraints:

$$\psi_k = X_{ck}\;C\;Y_{ck}^T \qquad k = 1, 2, 3, \ldots, N$$

where:

$$X_{ck} = [1, X_{ck}, X_{ck}^2, \ldots, X_{ck}^T]$$

Where X_{ck} is the X location of the k-th observation constraint, ψ_k .

$$Y_{ck} = [1, Y_{ck}, Y_{ck}^2, \ldots, Y_{ck}^U]$$

Where Y_{ck} is the Y location of the k-th observation constraint, ψ_k .

NOTE: The location of the constraints are not restricted to the orthonormal observation grid values!

utilizing the modified objective function, ϕ' :

$$\phi' = \left\| \Omega_x (W - X C Y^T) \Omega_y \right\| + \sum_{k=1}^{N} \lambda_k (\psi_k - X_{ck} C Y_{ck}^T)$$

where λ_k are the N Lagrange multipliers.

Minimizing this function gives:

$$\frac{\partial \phi'}{\partial C} = -2 X^T \Omega_x [W - X C Y^T] \Omega_y Y - \sum_{k=1}^{N} \lambda_k X_{ck}^T Y_{ck}$$

solving for C ;

$$-2 X^T \Omega_x W \Omega_y Y + 2 X^T \Omega_x X C Y^T \Omega_y Y - \sum_{k=1}^{N} \lambda_k C_{ck}^T Y_{ck} = 0$$

rearranging:

$$X^T \Omega_x X C Y^T \Omega_y Y = X^T \Omega_x W \Omega_y Y + \sum_{k=1}^{N} \frac{\lambda_k}{2} X^T_{ck} Y_{ck}$$

assuming the inverses exist:

$$C = (X^T \Omega_x X)^{-1} [X^T \Omega_x W \Omega_y Y + \sum_{k=1}^{N} \frac{\lambda_k}{2} X^T_{ck} Y_{ck}] (Y^T \Omega_y Y)^{-1}$$

and to evaluate the Lagrange multipliers:

$$\psi_k = X_{ck} C Y^T_{ck} \qquad k = 1, 2, 3, \ldots N$$

substituting:

$$\psi_k = X_{ck} (X^T \Omega_x X)^{-1} [X^T \Omega_x W \Omega_y Y + \sum_{j=1}^{N} \frac{\lambda_j}{2} X^T_{cj} Y_{cj}] (Y^T \Omega_y Y)^{-1} Y^T_{ck}$$

Since, by design, there are N constraint equations and N Lagrange multipliers, this set of equations represents a consistant, determined set that can be solved for the unknowns.

The closed-form matrix equation for the Lagrange multipliers in the more general case $N > 1$ has not been found as yet.

Consider the case when only one constraint is present; then, the constraint equation becomes

$$\psi = X_c \, (X^T \, \Omega_x \, X)^{-1} \, [X^T \, \Omega_x \, W \, \Omega_y \, Y] \, (Y^T \, \Omega_y \, Y)^{-1} \, Y_c^T$$
$$+ \lambda \, X_c \, (X^T \, \Omega_x X)^{-1} \, X_c^T \, Y_c \, (Y^T \, \Omega_y \, Y)^{-1} \, Y_c^T$$

or rearranging:

$$\lambda = \frac{\psi - X_c \, (X^T \, \Omega_x \, X)^{-1} \, [X^T \, \Omega_x \, W \, \Omega_y \, Y] \, (Y^T \, \Omega_y \, Y)^{-1} \, Y_c}{X_c \, (X^T \, \Omega_x \, X)^{-1} \, X_c^T \, Y_c \, (Y^T \, \Omega_y \, Y)^{-1} \, Y_c}$$

APPENDIX B. LEAST-SQUARES POLYNOMIAL FITTING

THE DERIVATION OF $C_L = (A^T A)^{-1} A^T Z_L$

Given a set of n elevation observations, $\sum_{i=1}^{n} z_i$, suppose that one wishes to fit the polynomial

$$f(x_i, y_i) = c_0 + c_1 x_i + c_2 y_i + c_3 x_i y_i$$

as closely as possible to these observations. An accepted "best" fit occurs when the sum of the squares of the vertical distances from the observations to the described surface is minimized. The distance from an elevation point to the described surface may be defined as

$$d = z_i - f(x_i, y_i)$$

$$d^2 = (z_i - f(x_i, y_i))^2$$

The set of n distance-squares is therefore

$$S = \sum_{i=1}^{n} d_i^2 = \sum_{i=1}^{n} (z_i - c_0 - c_1 x_i - c_2 y_i - c_3 x_i y_i)^2$$

S will be minimized when its derivative is equal to zero.

$$\left.\begin{aligned}
{}^{*}\ \frac{\delta S}{\delta c_0} &= -2\Sigma(z_i - c_0 - c_1x_i - c_2y_i - c_3x_iy_i) = 0 \\
\frac{\delta S}{\delta c_1} &= -2\Sigma[(z_i - c_0 - c_1x_i - c_2y_i - c_3x_iy_i)(x_i)] = 0 \\
\frac{\delta S}{\delta c_2} &= -2\Sigma[(z_i - c_0 - c_1x_i - c_2y_i - c_3x_iy_i)(y_i)] =0 \\
\frac{\delta S}{\delta c_3} &= -2\Sigma[(z_i - c_0 - c_1x_i - c_2y_i - c_3x_iy_i)(x_iy_i)] = 0
\end{aligned}\right\} \text{(B1)}$$

$$\left.\begin{aligned}
&c_0n + c_1 \Sigma x_i + c_2 \Sigma y_i + c_3 \Sigma x_iy_i = \Sigma z_i \\
&c_0\Sigma x_i + c_1 \Sigma x_i^2 + c_2 \Sigma x_iy_i + c_3 \Sigma x_i^2y_i = \Sigma x_iz_i \\
&c_0 \Sigma y_i + c_1 \Sigma x_i y_i + c_2 \Sigma y_i^2 + c_3 \Sigma x_i y_i^2 = \Sigma y_iz_i \\
&c_0 \Sigma x_iy_i + c_1 \Sigma x_i^2y_i + c_2 \Sigma x y_i^2 + c_3 \Sigma x_i^2y_i^2 = \Sigma x_iy_iz_i
\end{aligned}\right\} \text{(B2)}$$

---

*Hereafter the symbol Σ indicates $\sum_{i=1}^{n}$

Define A as an n by m matrix of polynomial terms, C_L as an m by 1 matrix of coefficients, and Z_L as an n by 1 matrix of observations.

Then,

$$A = \begin{bmatrix} 1 & x_1 & y_1 & x_1y_1 \\ 1 & x_2 & y_2 & x_2y_2 \\ 1 & x_3 & y_3 & x_3y_3 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_n & y_n & x_ny_n \end{bmatrix}$$

$$C_L = \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix}$$

$$Z_L \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_n \end{bmatrix}$$

The left side of equation (B2) can now be written as

$$\begin{bmatrix} 1 & 1 & 1 \ldots 1 \\ x_1 & x_2 & x_3 \ldots x_n \\ y_1 & y_2 & y_3 \ldots y_n \\ x_1y_1 & x_2y_2 & x_3y_3 \ldots x_ny_n \end{bmatrix} \begin{bmatrix} 1 & x_1 & y_1 & x_1y_1 \\ 1 & x_2 & y_2 & x_2y_2 \\ 1 & x_3 & y_3 & x_3y_3 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_n & y_n & x_ny_n \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix}$$

which in matrix notation is

$$A^TAC_L$$

The right side of equation (B2) can now be written as

$$\begin{bmatrix} 1 & 1 & 1 \ldots 1 \\ x_1 & x_2 & x_3 \ldots x_n \\ y_1 & y_2 & y_3 \ldots y_n \\ x_1y_1 & x_2y_2 & x_3y_3 \ldots x_ny_n \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_n \end{bmatrix}$$

which in matrix notation is

$$A^TZ.$$

Equation (B2) can now be rewritten as

$$A^TAC_L = A^TZ_L$$

Solving for C_L , multiply the left side by $(A^TA)^{-1}$

$$(A^TA)^{-1}(A^TA)C_L = (A^TA)^{-1}A^TZ_L$$

$$C_L = (A^TA)^{-1}\ A^TZ_L$$

<u>A NUMERICAL EXAMPLE OF THE LEAST-SQUARES TECHNIQUE</u>. Given nine observations on a 3 by 3 grid,

$z(0,0) = 100$

$z(0,1) = 110$

$z(0,2) = 112$

$z(1,0) = 118$

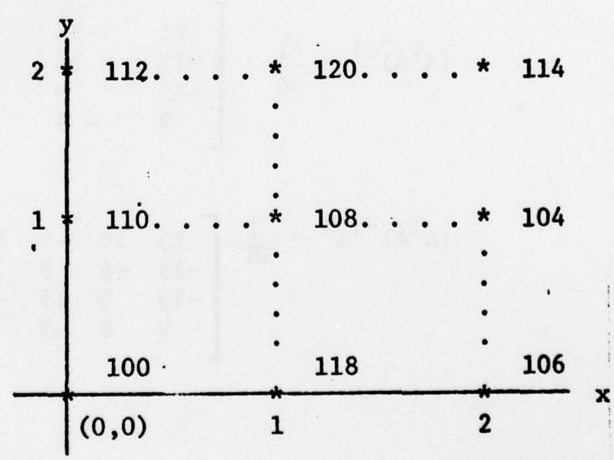
$z(1,1) = 108$

$z(1,2) = 120$

$z(2,0) = 106$

$z(2,1) = 104$

$z(2,2) = 114$



The model equation is chosen as

$$f(x,y) = c_0 + c_1x + c_xy + c_3xy$$

Then:

$$Z_L = \begin{bmatrix} 100 \\ 110 \\ 112 \\ 118 \\ 108 \\ 120 \\ 106 \\ 104 \\ 114 \end{bmatrix} \qquad A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 2 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 2 & 2 \\ 1 & 2 & 0 & 0 \\ 1 & 2 & 1 & 2 \\ 1 & 2 & 2 & 4 \end{bmatrix}$$

$$A^TA = \begin{bmatrix} 9 & 9 & 9 & 9 \\ 9 & 15 & 9 & 15 \\ 9 & 9 & 15 & 15 \\ 9 & 15 & 15 & 25 \end{bmatrix}$$

$$(A^TA)^{-1} = \frac{1}{36} \begin{bmatrix} 25 & -15 & -15 & 9 \\ -15 & 15 & 9 & -9 \\ -15 & 9 & 15 & -9 \\ 9 & -9 & -9 & 9 \end{bmatrix}$$

$$(A^TA)^{-1}A^T = \frac{1}{36} \begin{bmatrix} 25 & 10 & -5 & 10 & 4 & -2 & -5 & -2 & 1 \\ -15 & -6 & 3 & 0 & 0 & 0 & 15 & 6 & -3 \\ -15 & 0 & 15 & -6 & 0 & 6 & 3 & 0 & -3 \\ 9 & 0 & -9 & 0 & 0 & 0 & -9 & 0 & 9 \end{bmatrix}$$

$$(A^TA)^{-1}A^Tz_L = \frac{1}{36}\begin{bmatrix}3788\\48\\168\\-36\end{bmatrix} = \begin{bmatrix}105.222\\1.333\\4.667\\-1.000\end{bmatrix}$$

$c_0 = 105.222 \qquad c_1 = 1\ 1/3 \qquad c_2 = 4\ 2/3 \qquad c_3 = -1$

$$z = 105.222 + 1.333x + 4.667y - xy$$

Notice that this result is identical with that of the array algebra numerical example in appendix G.

APPENDIX C. ARRAY-ALGEBRA POLYNOMIAL FITTING

THE DERIVATION OF $C_A = (X^TX)^{-1}X^TZ_AY(Y^TY)^{-1}$

When the set of elevation observations

$$\sum_{i=0}^{m-1} z_i$$

is ordered in an orthonormal grid, then it may be written as

$$\sum_{i,j=0}^{(a-1),(b-1)} z_{ij}$$

with ab=m terms in the model equation. If the model equation is chosen as the four-term polynomial

$$z_{ij} = (x_i,y_i) = c_{00} + c_{10}x_i + c_{01}y_i + c_{11}x_iy_i$$

then,

$$z_{ij} = \sum_{k,l=0}^{1,1} \left(c_{kl}x_i^k y_j^l \right)$$

$$= \begin{bmatrix} 1 & x_i \end{bmatrix} \begin{bmatrix} c_{00} & c_{01} \\ c_{10} & c_{11} \end{bmatrix} \begin{bmatrix} 1 \\ y_j \end{bmatrix}$$

and, given n=ef observations, then all z_{ij} in the grid are

$$\begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ 1 & x_e \end{bmatrix} \begin{bmatrix} c_{00} & c_{01} \\ c_{10} & c_{11} \end{bmatrix} \begin{bmatrix} 1 & 1 & \cdots & 1 \\ y_1 & y_2 & \cdots & y_f \end{bmatrix}$$

$$= Z_A = XC_A Y^T$$

where

$$X = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ 1 & x_e \end{bmatrix}$$

$$C_A = \begin{bmatrix} c_{00} & c_{01} \\ c_{10} & c_{11} \end{bmatrix}$$

$$Y = \begin{bmatrix} 1 & y_1 \\ 1 & y_2 \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ 1 & y_f \end{bmatrix}$$

It is, of course, possible to generalize this derivation by selecting a general model polynomial.

Thus, choosing

$$z_{ij}=f(x_i y_j)=c_{00}+c_{10}x_i+c_{01}y_j+c_{11}x_i y_j+c_{20}x_i^2+c_{02}y_j^2$$

$$+c_{21}x_i^2 y_j+c_{12}x_i y_j^2+c_{22}x_i^2 y_j^2+\ldots+c_{(a-1)(b-1)}x_i^{(a-1)}y_j^{(b-1)}$$

then,

$$z_{ij}=\sum_{k,l=0}^{(a-1),(b-1)} c_{kl}x_i^k y_j^l$$

$$=\begin{bmatrix}1 & x_i & x_i^2 & x_i^3 \ldots x_i^{(a-1)}\end{bmatrix}\begin{bmatrix}c_{00} & c_{01} & c_{02} & \ldots & c_{0(b-1)}\\ c_{10} & c_{11} & c_{12} & \ldots & c_{1(b-1)}\\ c_{20} & c_{21} & c_{22} & \ldots & c_{2(b-1)}\\ \cdot & \cdot & \cdot & & \cdot\\ \cdot & \cdot & \cdot & & \cdot\\ \cdot & \cdot & \cdot & & \cdot\\ c_{(a-1)0} & c_{(a-1)1} & c_{(a-1)2} & \ldots & c_{(a-1)(b-1)}\end{bmatrix}\begin{bmatrix}1\\ y_j\\ y_j^2\\ \cdot\\ \cdot\\ \cdot\\ y_j^{(b-1)}\end{bmatrix}$$

and all z_{ij} in the grid are

$$
= \begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{(a-1)} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{(a-1)} \\ 1 & x_3 & x_3^3 & \cdots & x_3^{(a-1)} \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ 1 & x_e & x_e^2 & \cdots & x_e^{(a-1)} \end{bmatrix}
\begin{bmatrix} c_{00} & c_{01} & c_{02} & \cdots & c_{0(b-1)} \\ c_{10} & c_{11} & c_{12} & \cdots & c_{1(b-1)} \\ c_{20} & c_{21} & c_{22} & \cdots & c_{2(b-1)} \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ c_{(a-1)0} & c_{(a-1)1} & c_{(a-1)2} & \cdots & c_{(a-1)(b-1)} \end{bmatrix}
\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ y_1 & y_2 & y_3 & \cdots & y_f \\ y_1^2 & y_2^2 & y_3^2 & \cdots & y_f^2 \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ y_1^{(b-1)} & y_2^{(b-1)} & y_3^{(b-1)} & \cdots & y_f^{(b-1)} \end{bmatrix}
$$

$$= Z_A = XC_AY^T$$

Where X is an e by a matrix of x-direction parameters, Y is a f by b matrix of y-direction parameters, C_A is an a by b matrix of coefficients, and Z_A is an e by f matrix of elevation observations.

Solving for C_A

$$Z_A = XC_AY^T$$

$$Z_AY = X\,C_AY^TY$$

$$X^TZ_AY = (X^TX)C_A(Y^TY)$$

$$(X^TX)^{-1}\,X^TZ_AY = (X^TX)^{-1}(X^TX)C_A(Y^TY) = C_A(Y^TY)$$

$$(X^TX)^{-1}X^TZ_AY(Y^TY)^{-1} = C_A(Y^TY)(Y^TY)^{-1} = C_A$$

$$\underline{C_A = (X^TX)^{-1}X^TZ_AY(Y^TY)^{-1}}$$

Note that when X = Y, a special computational case exists:

$$Y(Y^TY)^{-1} = X(X^TX)^{-1}$$

$$= \left[[X({}^TX)^{-1}]^T \right]^T$$

$$= \left[[(X^TX)^{-1}]^T X^T \right]^T$$

and, since $(X^TX)^{-1}$ is symmetric,

$$\left[[(X^TX)^{-1}]^T X^T \right]^T = [(X^TX)^{-1} X^T]^T$$

therefore, when $X = Y$,

$$C_A = [(X^TX)^{-1} X^T] \, Z_A \, [(X^TX)^{-1} X^T]^T$$

and $Y(Y^TY)^{-1}$ need not be calculated.

A NUMERICAL EXAMPLE OF THE ARRAY ALGEBRA TECHNIQUE. The same observational values used in the numerical example in appendix A will also apply to this example.

The model equation $z = c_{00} + c_{01}x + c_{10}y + c_{11}xy$ can be written

$$z = (1,x) \begin{bmatrix} c_{00} & c_{01} \\ c_{10} & c_{11} \end{bmatrix} \begin{bmatrix} 1 \\ y \end{bmatrix}$$

therefore,

$$\begin{bmatrix} c_{00} & c_{01} \\ c_{10} & c_{11} \end{bmatrix} = (X^TX)^{-1} X^TZ_AY(Y^TY)^{-1}$$

where $X = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{bmatrix}$ $\quad Y = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{bmatrix}$

and $$Z_A = \begin{bmatrix} z_{00} & z_{01} & z_{02} \\ z_{10} & z_{11} & z_{12} \\ z_{20} & z_{21} & z_{22} \end{bmatrix} = \begin{bmatrix} 100 & 110 & 112 \\ 118 & 108 & 120 \\ 106 & 104 & 114 \end{bmatrix}$$

$$(X^TX) = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 2 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{bmatrix} = \begin{bmatrix} 3 & 3 \\ 3 & 5 \end{bmatrix}$$

$$(X^TX)^{-1} = \frac{1}{6} \begin{bmatrix} 5 & -3 \\ -3 & 3 \end{bmatrix}$$

$$(X^TX)^{-1}X^T = \frac{1}{6} \begin{bmatrix} 5 & -3 \\ -3 & 5 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 2 \end{bmatrix} = \frac{1}{6} \begin{bmatrix} 5 & 2 & -1 \\ -3 & 0 & 3 \end{bmatrix}$$

since $Y = X$, $Y(Y^TY)^{-1} = [(X^TX)^{-1}X^T]^T$

$$Y(Y^TY)^{-1} = \frac{1}{6} \begin{bmatrix} 5 & -3 \\ 2 & 0 \\ -1 & 3 \end{bmatrix}$$

$$C_A = \frac{1}{6} \begin{bmatrix} 5 & 2 & -1 \\ -3 & 0 & 3 \end{bmatrix} \begin{bmatrix} 100 & 110 & 112 \\ 118 & 108 & 120 \\ 106 & 104 & 114 \end{bmatrix} \frac{1}{6} \begin{bmatrix} 5 & -3 \\ 2 & 0 \\ -1 & 3 \end{bmatrix}$$

$$= \frac{1}{36} \begin{bmatrix} 630 & 662 & 686 \\ 18 & -18 & 6 \end{bmatrix} \begin{bmatrix} 5 & -3 \\ 2 & 0 \\ -1 & 3 \end{bmatrix}$$

$$= \frac{1}{36} \begin{bmatrix} 3788 & 168 \\ 48 & -36 \end{bmatrix}$$

$$= \begin{bmatrix} 105.222 & 4\ 2/3 \\ 4/3 & -1 \end{bmatrix}$$

$$Z = 105.222 + 1.333x + 4.667y - xy$$

Notice that this result is identical with that of the least-squares numerical example in appendix B.

APPENDIX D. THE NUMBER OF MULTIPLICATIONS AND ADDITIONS REQUIRED TO FIT A POLYNOMIAL OF m TERMS TO DATA SETS OF n OBSERVATIONS

REQUIRED MULTIPLICATIONS AND ADDITIONS USING THE METHOD OF LEAST-SQUARES.

Given that

$$C_L = (A^T A)^{-1} A^T \; Z_L$$

where

A is an nxm matrix, Z_L is an nx1 matrix.

Then

$L = A^T A$ requires

$$n\sum_{i=1}^{m} i \quad \text{multiplications} \quad \text{and} \quad (n-1)\sum_{i=1}^{m} i \quad \text{additions}$$

which simplify to

$$\frac{1}{2}\, nm(m+1) \quad \text{multiplications} \quad \text{and} \quad \frac{1}{2}\, m(n-1)(m+1) \text{ additions.}$$

$P = (L)^{-1}$ requires

$$\frac{1}{6}\, m(4m^2+3m-1) \quad \text{multiplications} \quad \text{(see reference 13)}$$

and

$$\frac{1}{6}\, m(4m^2-3m-1) \quad \text{additions} \quad \text{(see reference 13)}$$

$Q = PA^T$ requires

m^2n multiplications and $mn(m-1)$ additions.

$R = (A^TA)^{-1}A^T$ requires

$\frac{1}{6}\, m(4m^2 + 3m + 9mn + 3n - 1)$ multiplications

and

$\frac{1}{6}\, m(4m^2 - 6m + 9mn - 3n - 4)$ additions.

In the polynomial fitting process, it is necessary to compute $(A^TA)^{-1}A^T$ only once. The multiplication of $(A^TA)^{-1}A^T$ to Z_L, however, must be repeated as many times as necessary to compute all of the polynomials needed to describe the terrain.

This process requires

(mn)(# of reps.) multiplications, and

m(n-1)(# of reps.) additions.

REQUIRED MULTIPLICATIONS AND ADDITIONS USING ARRAY ALGEBRA.

$$C_A = (X^TX)^{-1}X^T \; Z_A \; Y(Y^TY)^{-1}$$

where

X is an e x a matrix

Y is an f x b matrix

ab = m(terms in the model equation)

X^TX requires

$\frac{1}{2}ae(a + 1)$ multiplications

and

$\frac{1}{2}a(e - 1)(a + 1)$ additions.

$(X^TX)^{-1}$ requires

$\frac{1}{6}a(4a^2 + 3a - 1)$ multiplications (see reference 13)

and

$\frac{1}{6}a(4a^2 - 3a - 1)$ additions (see reference 13)

$(X^TX)^{-1}X^T$ requires

a^2e multiplications

and

$ae(a - 1)$ additions.

$(X^TX)^{-1}X^T$ therefore requires

$\frac{1}{6}a(4a^2 + 3a + 9ae + 3e - 1)$ multiplications

and

$$\frac{1}{6}\, a(4a^2 - 6a + 9ae - 3e - 4) \qquad \text{additions.}$$

$Y(Y^TY)^{-1}$ likewise requires

$$\frac{1}{6}\, b(4b^2 + 3b + 9bf + 3f - 1) \qquad \text{multiplications}$$

and

$$\frac{1}{6}\, b(4b^2 - 6b + 9bf - 3f - 4) \qquad \text{additions.}$$

The repetitive computations involve the multiplication of $(X^TX)^{-1}X^T$ to Z and the multiplication of $(X^TX)^{-1}X^TZ$ to $Y(Y^TY)^{-1}$. These steps require

$$p(an + mf) \qquad \text{multiplications}$$

and

$$p(an - af + mf - m) \qquad \text{additions}$$

where p is equal to the number of polynomial fits per map sheet.

Where $X = Y$,

$$Y(Y^TY)^{-1} = X(X^TX)^{-1}$$

$$= \left([X(X^TX)^{-1}]^T\right)^T$$

which, since $(X^TX)^{-1}$ is symmetric, is equal to $[(X^TX)^{-1}X^T]^T$

therefore, $Y(Y^TY)^{-1}$ need not be calculated.

APPENDIX E. POLYNOMIAL COVERAGE AND THE NUMBER OF POLYNOMIALS PER MAPSHEET

Reference two presents empirical evidence that the best approximation of automatic compilation elevation data for a 1:50,000 output scale and a density of approximately one million points per mapsheet is achieved for a roving, locally valid, four coefficient polynomial model equation when it is calculated from $169 = 13^2$ data points using 46% overlap. For the purpose of studying larger model equations it was assumed that each coefficient added would allow for a proportional increase in area covered, e.g., the number of terms in the model equation, m , would be related to the number of observations used to compute it by the formula:

$$n = (\text{INTEGER } \sqrt{42.25m})^2$$

which is derived directly from the case in point.

Using the 46% overlap and a typical $916 \times 1112 \approx 10^6$ data grid for a 1:50,000 mapsheet, the number of polynomials, p , needed per map sheet is given by:

$$p = [\text{INTEGER ROUNDED UP } \frac{916}{\frac{1}{2}(e+1)}][\text{INTEGER ROUNDED UP } \frac{1112}{\frac{1}{2}(f+1)}]$$

where $e = f = \sqrt{n}$ for unbiased modeling. The denominators compute the point spacings between neighboring local polynomial's centers. The two factors used then compute the number of rows and columns of polynomials, respectively. Note that by successive use of these formulae, the effect of larger polynomial model equations resulting in fewer polynomials per sheet is automatically accounted for.

D—79,864—AG—Ft Belvoir