ANTENNA LABORATORY REPORT NO. 72-11

A MATRIX EQUATION SOLUTION BY AN OPTIMIZATION TECHNIQUE

BY

M. J. JOHNSON

AND

R. MITTRA

SCIENTIFIC REPORT NO. 20

DECEMBER 1972

SPONSORED BY

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

NGR-14-005-009

ANTENNA LABORATORY
DEPARTMENT OF ELECTRICAL ENGINEERING
ENGINEERING EXPERIMENT STATION
UNIVERSITY OF ILLINOIS
URBANA, ILLINOIS 61801

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Department of Electrical Engineering University of Illinois Urbana, Illinois 61801	2a. REPORT SECURITY CLASSIFICATION 2b. GROUP

3. REPORT TITLE
A MATRIX EQUATION SOLUTION BY AN OPTIMIZATION TECHNIQUE

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)
Scientific-December 1972

5. AUTHOR(S) (First name, middle initial, last name)
Martin Johnson
Raj Mittra

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
December 1972	48	5

8a. CONTRACT OR GRANT NO NGR-14-005-009 b. PROJECT NO. c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) Antenna Laboratory Report No. 72-11 Technical Report No. 20 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) UILU-ENG-72-2557

10. DISTRIBUTION STATEMENT
Unlimited

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	National Aeronautics and Space Administration

13. ABSTRACT

The computer solution of matrix equations is often difficult to accomplish due to an ill-conditioned matrix or high noise levels. This paper compares two methods of solution for matrices of various degrees of ill-conditioning and for various noise levels in the right-hand side vector. One method employs the usual Gaussian elimination. The other solves the equation by an optimization technique and employs a function minimization subroutine.

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Matrix						
Optimization						
Noise						
Ill-Conditioned						

UILU-72-2557

ANTENNA LABORATORY REPORT NO. 72-11

A MATRIX EQUATION SOLUTION BY AN OPTIMIZATION TECHNIQUE

BY

M. J. JOHNSON

AND

R. MITTRA

SCIENTIFIC REPORT NO. 20

DECEMBER 1972

SPONSORED BY

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

NGR-14-005-009

ANTENNA LABORATORY
DEPARTMENT OF ELECTRICAL ENGINEERING
ENGINEERING EXPERIMENT STATION
UNIVERSITY OF ILLINOIS
URBANA, ILLINOIS 61801

ABSTRACT

The computer solution of matrix equations is often difficult to accomplish due to an ill-conditioned matrix or high noise levels. This paper compares two methods of solution for matrices of various degrees of ill-conditioning and for various noise levels in the right-hand side vector. One method employs the usual Gaussian elimination. The other solves the equation by an optimization technique and employs a function minimization subroutine.

TABLE OF CONTENTS

✓

1. INTRODUCTION

It is a common occurrence in engineering and science to find the equation involving an unknown function expressed as an equality where the left-hand side consists of an operator operating on the unknown function and the right-hand side consists of a known function. Frequently, if the known and unknown functions have limited domains, then the method of moments may be used to express both functions as vectors and the operator as a matrix. The problem of solving for the unknown function then is reduced to a linear algebraic problem.

Frequently, the right-hand side function is a measured quantity and therefore can be expected to contain some amount of noise. This paper compares two methods for solving a matrix equation with various amounts of noise in the right-hand side vector. The first method is the most commonly used and employs Gaussian elimination.. The goal at the outset of this project was to find a method which could solve a matrix equation of higher condition number and greater noise level than previously possible using a Gaussian elimination-type subroutine. The second method uses an optimization technique, wherein the magnitude of the difference between the right- and left-hand sides is minimized.

A matrix was found whose condition number could be systematically varied. At the same time varying amounts of noise were introduced into the right-hand side vector. The above two methods were compared, and finally two illustrative problems were solved using the optimization technique.

2. THE TEST MATRIX

To test the two methods, a matrix whose condition number could be systematically varied was necessary. The condition number used in this section is given by

$$\text{condition number} = \frac{|\lambda|_{max}}{|\lambda|_{min}} \quad .$$

A matrix whose off-diagonal elements are all equal to one and whose diagonal elements are equal to one plus delta has a simple expression for the condition number. It is first necessary to derive a formula for the determinant of such a matrix.

For an N × N matrix called A, consider

$$J_N = |A_N| = \begin{vmatrix} 1+\delta & 1 & \cdot & \cdot & 1 \\ 1 & 1+\delta & & & \cdot \\ \cdot & & \cdot & & \cdot \\ \cdot & & & \cdot & \cdot \\ 1 & \cdot & \cdot & \cdot & 1+\delta \end{vmatrix}$$

where N stands for the dimension. Expanding J_N by co-factors we obtain

$$J_N = (1 + \delta)J_{N-1} - K_2 + K_3 - K_4 \; . \; . \; . \; K_N$$

where K_j is equal to the minor which results after deleting the j-th column and first row. However, each K_{j-1} is just equal to $-K_j$ since a single row interchange converts K_{j+1} to K_j. Thus, except for sign changes, all the K's are equal. Letting $K_2 = J'_{N-1}$ we obtain

$$J'_{N-1} = K_2 = \begin{vmatrix} 1 & 1 & \cdot & \cdot & 1 \\ 1 & 1+\delta & & & \cdot \\ \cdot & & \cdot & & \cdot \\ \cdot & & & \cdot & \cdot \\ 1 & \cdot & \cdot & \cdot & 1+\delta \end{vmatrix} \quad .$$

It can be shown by induction that the formula is

$$J_N = \delta^N + N\delta^{N-1} \quad . \tag{1}$$

We now use this result to find the eigenvalues of the matrix A. The eigenvalue equation is

$$|A - \lambda I| = 0 \quad . \tag{2}$$

Using Equations (1) and (2), we get

$$(\delta - \lambda)^N + N(\delta - \lambda)^{N-1} = 0$$

$$\lambda_1, \lambda_2, \; . \; . \; . \; \lambda_{N-1} = \lambda_{min.} = \delta$$

$$\lambda_N = \lambda_{max.} = \delta + N$$

$$\text{condition number} = \frac{|\lambda|_{max.}}{|\lambda|_{min.}}$$

$$= \frac{\delta + N}{\delta}$$

$$\text{condition number} = 1 + \frac{N}{\delta} \quad . \tag{3}$$

We are now able to construct a matrix of desired condition number merely by choosing the proper δ and N. We next test the two methods for solving matrix equations.

3. COMPARISON OF SUBROUTINES

The first method makes use of DGELG, an IBM SSP subroutine, modified to include complex numbers. DGELG solves the matrix equation by means of Gaussian elimination. Let A be the matrix, x the unknown vector, and y the known vector. Then to solve

$$Ax = y$$

for x, one merely supplies DGELG with A and y, and DGELG returns x.

The second method attempts to find x by minimizing the square magnitude of $Ax - y$. That is, minimize $h(x)$ where

$$h(x) = (Ax - y)^{\dagger} (Ax - y) \tag{4}$$

and † signifies Hermitian adjoint. If we let the superscripts R and I stand for real and imaginary parts respectively, then Equation (4) expands into

$$\begin{aligned} h &= |(Ax - y)^R|^2 + |(Ax - y)^I|^2 \\ &= [(Ax - y)^R]^T (Ax - y)^R + [(Ax - y)^I]^T (Ax - y)^I \\ &= |(A^R x^R - A^I x^I - y^R)|^2 + |(A^I x^R + A^R x^I - y^I)|^2 \\ &= \left| \begin{pmatrix} A^R & -A^I \\ A^I & A^R \end{pmatrix} \begin{pmatrix} x^R \\ x^I \end{pmatrix} - \begin{pmatrix} y^R \\ y^I \end{pmatrix} \right|^2 . \end{aligned}$$

Thus the N-dimensional complex problem has been reduced to a 2N-dimensional real problem.

A', x' and y' will be defined as follows

$$A' = \begin{pmatrix} A^R & -A^I \\ A^I & A^R \end{pmatrix}$$

$$x' = \begin{pmatrix} x^R \\ x^I \end{pmatrix}$$

$$y' = \begin{pmatrix} y^R \\ y^I \end{pmatrix} \quad .$$

Then,

$$h = (A'x' - y')^T \, (A'x' - y')$$

where T signifies the transpose. Expanding h, we get

$$\begin{aligned} h &= (x'^T A'^T - y'^T) \, (A'x' - y') \\ &= x'^T A'^T A'x' - y'^T A'x' - x'^T A'^T y' + y'^T y' \\ &= x'A'^T A'x' - 2x'^T A'^T y' + y'^T y' \\ \nabla h &= 2A'^T A'x' - 2A'^T y' \\ &= 2(A'^T A'x' - A'^T y') \quad . \end{aligned}$$

We now have explicit formulas for the function value and its gradient vector. The IBM SSP subroutine DFMFP was chosen to minimize this function. DFMFP uses a variable metric version of a conjugate gradient algorithm devised by Fletcher and Powell.[1]

The program used consists of one main and four subroutines. The first subroutine, OPER, constructs the matrix A' and the vector y' with varying amounts of noise. To construct a specific y', x' was chosen to be

$$x' = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} .$$

The problem, then, is to recover this x', given A' and y'.

The subroutine PRELIM constructed $A'^T A'$ and $A'^T Y'$ referred to as ATA and ATY, respectively, in PRELIM. The subroutine FUNCT supplied the function value h(x) and the gradient vector for a given x. This method requires an initial guess be given for x'. It was chosen such that each element of x' was equal to one-half.

Random noise at percentage levels of from 0.0 to 2.25 percent was added to the y' vector on successive iterations by OPER. The noise was supplied by a random number generator yielding numbers between zero and one in an even distribution.

If the magnitude of the vector, equal to the difference between the calculated x' vector and the exact x' vector, was less than 10 percent of the magnitude of the exact x' vector, the solution was considered successful. The results of the tests are tabulated in Table 3.1. The condition numbers were computed from Equation (3).

Under no-noise conditions, DGELG was able to solve the equation up to the point where the condition number became equal to 10^{15}. At this point the diagonal elements became equal to 1.0 plus 10^{-15} which contains the maximum number of significant digits that the machine can hold.

TABLE 3.1

PERCENTAGE ERROR BETWEEN THE CALCULATED AND THE TRUE X VECTORS FOR GAUSSIAN AND OPTIMIZATION TECHNIQUES UNDER NOISY CONDITIONS. G INDICATES GAUSSIAN. O INDICATES OPTIMIZATION.

Condition Number	Percent Noise 0.0	0.25	0.50	0.75	1.00	1.25	1.50	1.75	2.00	2.25
5.0×10^0	0.0 G 0.0 O	.3 G .4 O	1.1 G .9 O	.9 G .9 O	2.1 G 2.1 O	1.4 G 2.0 O	3.2 G 3.8 O	2.4 G 3.3 O	3.7 G 2.2 O	5.1 G 4.8 O
1.7×10^1	0.0 G 0.0 O	..8 G .6 O	4.7 G 2.0 O	4.7 G 2.4 O	3.6 G 6.9 O	3.8 G 4.9 O	7.2 G		6.1 O	
6.5×10^1	0.0 G 0.0 O	4.7 G 3.0 O	6.7 O							
2.6×10^2	0.0 G 0.0 O									
4.1×10^3	0.0 G 0.0 O									
1.0×10^4	0.0 G 0.0 O									
1.0×10^5	0.0 G 0.0 O									
1.0×10^6	0.0 G 0.0 O									
1.0×10^7	0.0 G 0.0 O									
.										
.										
1.0×10^{15}	0.0 G 0.0 O									

$$\text{Percent Error} = \frac{|x_c - x|}{|x|} \quad \text{where } x_c = \text{the calculated } x, \quad x = \text{the true } x$$

Omission implies error > 10 percent

The computer cannot hold a more ill-conditioned matrix of the form in question.

For large numbers the function value and gradient had to be multiplied by a large number within FUNCT. If this number was too large, overflow resulted within DFMFP. If this number was too small, underflow resulted. The number which gave the best results was 10^{17}. The results of both methods are summarized in Table 3.1.

4. FIRST EXAMPLE

In this example a mode-matching technique is employed to solve a bifurcated, infinite, parallel-plane, waveguide problem. The waveguide in question is pictured in Figure (1). If a $TE_{1,0}$ wave is incident from the left, then the resulting fields can be determined by first solving the wave equation for the infinite number of normal modes in each region and then matching at $z = 0$. This results in an infinite number of linear equations in an infinite number of unknowns. The solution may be approximated by truncating the coefficient matrix and both the known and unknown vectors. The elements of the unknown vector correspond to the amplitudes of the normal modes in region A. A more complete discussion of the problem and the exact formulas for the matrix and vector elements may be found in Analytical Techniques in the Theory of Guided Waves.[2]

For this example, the matrix was limited to 25 by 25. The distance between the outermost conductors was made equal to 25 units. The problem was solved for a center conductor located at distances from 10 to 18 units from the top conductor. In the matrix generating program, the parameter p is equal to this distance. The subroutine generating the truncated matrix and known vector was written by C. A. Klein,[3] who attempted a solution by Gaussian elimination. His results are tabulated in Tables 4.1 through 4.5.

The same problem was solved using the optimization technique with 0.0 percent and 5.0 percent noise introduced into the known vector. Klein also computed the condition numbers by using the new definition

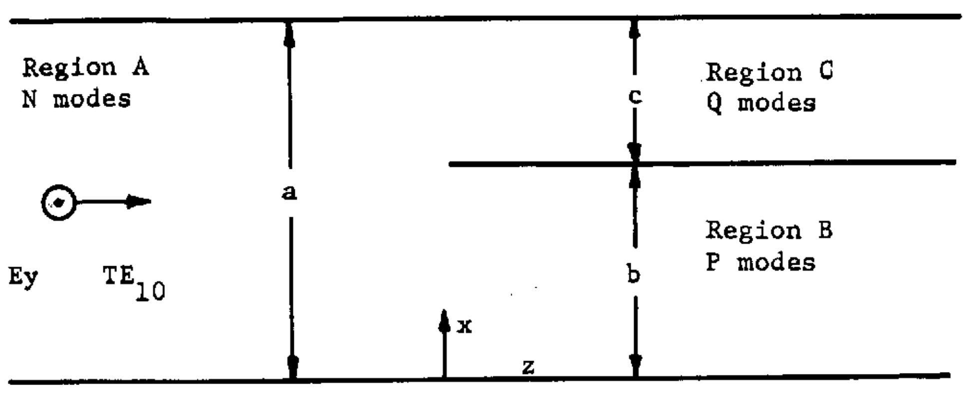

Figure 1. Bifurcated waveguide

TABLE 4.1. NORMAL MODE AMPLITUDES FOR BIFURCATED WAVEGUIDE

p = 10

% Noise = 0.0

Method - Gauss

Real(x)	Imaginary(x)
-0.693152707549	-0.720790762992
0.155337724329	-0.364891591341
0.317913267895D-01	-0.746784972755D-01
-0.733016799930D-01	0.172187192624
0.303396945421D-01	-0.712685825042D-01
0.303679488886D-01	-0.713349525603D-01
-0.484276679557D-01	0.113757613624
0.109584438701D-01	-0.257416158227D-01
0.408810908144D-01	-0.960305447226D-01
-0.494418659956D-01	0.116139986216
-0.543902557502D-02	0.127763858138D-01
0.778883268731D-01	-0.182961322905
-0.776280356439D-01	0.182349893318
-0.660471391558D-01	0.155146123165
0.319067252346	-0.749495706601
-0.354665663973	0.833117126476
-3.63453198601	8.53759230698
-8.28472689191	19.4609982387
-10.9104185003	25.6288032169
-9.70015620899	22.7858715639
-6.10127601639	14.3320260716
-2.71727834179	6.38294414712
-0.825048941140	1.93805736755
-0.154679367040	0.363345096218
-0.136187070269D-01	0.319906301001D-01

TABLE 4.2. NORMAL MODE AMPLITUDES FOR BIFURCATED WAVEGUIDE

p = 12

% Noise = 0.0

Method - Gauss

Real(x)	Imaginary(x)
-0.612978876221	-0.790099295853
0.158265570419	-0.323097392011
0.297182792634D-01	-0.606695347549D-01
-0.628349839686D-01	0.128276917042
0.236976248805D-01	-0.483784361655D-01
0.214179469823D-01	-0.437244992318D-01
-0.304892338095D-01	0.622434298386D-01
0.607189332784D-02	-0.123957023223D-01
0.195858816160D-01	-0.399843582757D-01
-0.200289933433D-01	0.408889659114D-01
-0.181000234917D-02	0.369509955324D-02
0.204861632204D-01	-0.418222731026D-01
-0.152907869413D-01	0.312159705326D-01
-0.899164794132D-02	0.183563487120D-01
0.263041303798D-01	-0.536995879920D-01
-0.136079851794D-01	0.277805495556D-01
-0.260088498294D-01	0.530967760506D-01
0.578118299740D-01	-0.118022204340
-0.164515724349D-01	0.335857011358D-01
-0.387550833889	0.791180693118
-0.720974673427	1.47186173261
-0.677714093308	1.38354574211
-0.372769785929	0.761005349015
-0.116096347718	0.237009395475
-0.160859761550D-01	0.328393404191D-01

TABLE 4.3. NORMAL MODE AMPLITUDES FOR BIFURCATED WAVEGUIDE

p = 14

% Noise = 0.0

Method - Gauss

Real(x)	Imaginary(x)
-0.562347051281	-0.826901320543
0.158937293771	-0.300296064468
0.283805678513D-01	-0.536222344729D-01
-0.570667538155D-01	0.107821903696
0.204014765122D-01	-0.385465422279D-01
0.173978355082D-01	-0.328714640184D-01
-0.232337332473D-01	0.438978070629D-01
0.431047696895D-02	-0.814421360174D-02
0.128448798674D-01	-0.242691113031D-01
-0.120121195993D-01	0.226956943585D-01
-0.980412231164D-03	0.185239050942D-02
0.986816122027D-02	-0.186449001846D-01
-0.642276774098D-02	0.121351750105D-01
-0.321036596398D-02	0.606566427305D-02
0.771365634861D-02	-0.145741794716D-01
-0.312358858949D-02	0.590170713358D-02
-0.434970895889D-02	0.821833850912D-02
0.626398205501D-02	-0.118351653937D-01
-0.919874992913D-03	0.173801147370D-02
-0.564189590637D-02	0.106597960530D-01
0.618814322022D-02	-0.116918755271D-01
0.157946753778D-02	-0.298424538569D-02
-0.190921166762D-01	0.360726382347D-01
-0.210777001712D-01	0.398241989608D-01
-0.706265865991D-02	0.133441846777D-01

TABLE 4.4. NORMAL MODE AMPLITUDES FOR BIFURCATED WAVEGUIDE

p = 16

% Noise = 0.0

Method - Gauss

Real(x)	Imaginary(x)
-0.544655824876	-0.838659664243
0.159002204125	-0.292852620963
0.279127637618D-01	-0.514101428403D-01
-0.551811109372D-01	0.101633389641
0.193746450077D-01	-0.356845089159D-01
0.162021930016D-01	-0.298414396956D-01
-0.211782419486D-01	0.390064005475D-01
0.383719356428D-02	-0.706740009441D-02
0.111371878738D-01	-0.205126380288D-01
-0.101119925885D-01	0.186244181268D-01
-0.798234279886D-03	0.147019975159D-02
0.773455715437D-02	-0.142456222359D-01
-0.481838950870D-02	0.887458135696D-02
-0.228870469474D-02	0.421537029726D-02
0.517804760418D-02	-0.953700497864D-02
-0.195088330013D-02	0.359316583554D-02
-0.248733502877D-02	0.458121059641D-02
0.320720168505D-02	-0.590707169496D-02
-0.408219513817D-03	0.751864763181D-03
-0.206342688285D-02	0.380045027762D-02
0.171162009022D-02	-0.315248730212D-02
0.278188244359D-03	-0.512371239947D-03
-0.129949453829D-02	0.239342833995D-02
0.619034571249D-03	-0.114014706686D-02
0.269179439590D-03	-0.495778689530D-03

TABLE 4.5. NORMAL MODE AMPLITUDES FOR BIFURCATED WAVEGUIDE

p = 18

% Noise = 0.0

Method - Gauss

Real(x)	Imaginary(x)
-0.565127177149	-0.825003802202
0.158936666466	-0.301521151135
0.284631279896D-01	-0.539978301243D-01
-0.574064797457D-01	0.108906700011
0.205920971239D-01	-0.390655785550D-01
0.176275737892D-01	-0.334415365493D-01
-0.236441667722D-01	0.448557059949D-01
0.440910063148D-02	-0.836457141979D-02
0.132180244197D-01	-0.250761138220D-01
-0.124499163053D-01	0.236189243136D-01
-0.102496495929D-02	0.194447650923D-02
0.104264309693D-01	-0.197801397122D-01
-0.687634469470D-02	0.130452174067D-01
-0.349539382475D-02	0.663116443264D-02
0.858518155290D-02	-0.162870776272D-01
-0.358117554900D-02	0.679390223769D-02
-0.519935462265D-02	0.986377420543D-02
0.796696919662D-02	-0.151142576263D-01
-0.129329530044D-02	0.245353005330D-02
-0.954607825673D-02	0.181100092811D-01
0.164964373311D-01	-0.312956404857D-01
-0.158733478035D-01	0.301135679293D-01
0.101404606956D-01	-0.192376211857D-01
-0.418116994558D-02	0.793216067202D-02
0.882768044447D-03	-0.167471259380D-02

$$\text{CONDITION NUMBER} = \frac{\sum_{i=1}^{25} |A_{ij}| \text{ max.}}{\sum_{i=1}^{25} |A_{ij}^{-1}| \text{ max.}} \quad .$$

The matrix was best conditioned at $p = 16$.

Under no-noise conditions and for p greater than 10, both methods obtained vectors whose respective elements were within 10 percent of each other. This was also true for p taking on values from fourteen through eighteen with 5 percent noise. In the remaining noisy cases, the optimization method was unable to get good results. The resulting data using the optimization method appears in Tables 4.6 through 4.15.

TABLE 4.6. NORMAL MODE AMPLITUDES FOR BIFURCATED WAVEGUIDE

p = 10

% Noise = 0.0

Method - Optimization

Real(x)	Imaginary(x)
-0.031659023411713160000000	0.070262998357441300000000
0.130983830676876900000000	-0.290829274906558100000000
-0.118873989529130900000000	0.264121874556566400000000
-0.916987505976385900000000	2.039898407327696000000000
-1.331309257116434000000000	2.969826447278865000000000
-0.642384359557458300000000	1.451867205489121000000000
0.533417061874102900000000	-1.154482551562311000000000
1.117930957716914000000000	-2.458386565041257000000000
0.893445763142052700000000	-1.970666473565769000000000
0.408868593328035000000000	-0.903006150710929000000000
0.106522300075213600000000	-0.235432568340416700000000
0.012462432602105240000000	-0.027557587897904990000000
-0.661633240792600900000000	-0.749798972961606800000000
0.156832939260419200000000	-0.347572358523052400000000
0.031005996720551770000000	-0.068718647591629500000000
-0.069056352160343700000000	0.153057321839261600000000
0.275442248865725300000000	-0.061052680367360000000000
0.026479186551324710000000	-0.058695594582244900000000
-0.040385459160621330000000	0.089527488529071100000000
0.008695464242043633000000	-0.019277892191076600000000
0.030673220303335630000000	-0.068009226448130300000000
-0.034807825322515870000000	0.077185426629525900000000
-0.003558463486326956000000	0.007891908114104444000000
0.046776694854617330000000	-0.103758971959666700000000
-0.042109717172825660000000	0.093428125539767900000000

TABLE 4.7. NORMAL MODE AMPLITUDES FOR BIFURCATED WAVEGUIDE

p = 12

% Noise = 0.0

Method - Optimization

Real(x)	Imaginary(x)
-0.61297892999733830000000000	-0.79009925697595220000000000
0.15826556989139770000000000	-0.32309741645946340000000000
0.02971828074969056000000000	-0.06066954238444657000000000
-0.06283499048815300000000000	0.12827693949065850000000000
0.02369762865233541000000000	-0.04837844709155629000000000
0.02141795161238888000000000	-0.04372451148114009000000000
-0.03048924223823323000000000	0.06224345083334724000000000
0.00607189547348921400000000	-0.01239570724573590000000000
0.01958588980553055000000000	-0.03998437691105665000000000
-0.02002900325433833000000000	0.04088898790683517000000000
-0.00181000341320027400000000	0.00369510184536166600000000
0.02048617710293876000000000	-0.04182230261125592000000000
-0.01529079885565838000000000	0.03121599550342821000000000
-0.00899165593294153500000000	0.01835636530599572000000000
0.02630415706095524000000000	-0.05369964280856439000000000
-0.01360800091935575000000000	0.02778058154494646000000000
-0.02600888409325488000000000	0.05309684502264355000000000
0.05781191650544647000000000	-0.11802237731324690000000000
-0.01645160061352412000000000	0.03358575686561898000000000
-0.38755158520168580000000000	0.79118217717903420000000000
-0.72097627127820640000000000	1.47186487267867200000000000
-0.67771581743233450000000000	1.38354911564497700000000000
-0.37277088045493610000000000	0.76100748287518520000000000
-0.11609674353493110000000000	0.23701016535098740000000000
-0.01608604005991786000000000	0.03283946476147687000000000

TABLE 4.8. NORMAL MODE AMPLITUDES FOR BIFURCATED WAVEGUIDE

p = 14

% Noise = 0.0

Method - Optimization

Real(x)	Imaginary(x)
-0.56234705128163460000000	-0.82690132054243560000000
0.15893729377109530000000	-0.30029606446610100000000
0.02838056785202951000000	-0.05362223447404557000000
-0.05706675381527905000000	0.10782190369524180000000
0.02040147651855036000000	-0.03854654223322561000000
0.01739783551014692000000	-0.03287146402163198000000
-0.02323373324606561000000	0.04389780706510034000000
0.00431047696920258300000	-0.00814421360206109400000
0.01284487986705568000000	-0.02426911130472287000000
-0.01201211959830290000000	0.02269569436104084000000
-0.00098041223121314090000	0.00185239050893449300000
0.00986816121973145900000	-0.01864490018591185000000
-0.00642276774030048400000	0.01213517501140921000000
-0.00321036596430562900000	0.00606566427249947800000
0.00771365634891726600000	-0.01457417947185004000000
-0.00312358858955307200000	0.00590170713457210300000
-0.00434970895941240800000	0.00821833850811271600000
0.00626398205579066800000	-0.01183516539258885000000
-0.00091987499285718850000	0.00173801147436736200000
-0.00564189590763467200000	0.01065979604938364000000
0.00618814322164534500000	-0.01169187552327318000000
0.00157946753727410100000	-0.00298424538670592900000
-0.01909211667493971000000	0.03607263823641620000000
-0.02107770017088156000000	0.03982419896024727000000
-0.00706265865980770200000	0.01334418467762105000000

TABLE 4.9. NORMAL MODE AMPLITUDES FOR BIFURCATED WAVEGUIDE

p = 16

% Noise = 0.0

Method - Optimization

Real(x)	Imaginary(x)
-0.544655824894647700000000	-0.838659664280518800000000
0.159002204090382300000000	-0.292852620749061100000000
0.027912763871352510000000	-0.051410142982278900000000
-0.055181110012006300000000	0.101633389433528600000000
0.019374645070393910000000	-0.035684508790109160000000
0.016202192913674400000000	-0.029841439940676720000000
-0.021178241651242110000000	0.039006401339599830000000
0.003837193482641438000000	-0.007067400183515707000000
0.011137187836275580000000	-0.020512638023302040000000
-0.010111992408212390000000	0.018624418384344160000000
-0.000798234381924184300000	0.001470199678759867000000
0.007734557265012296000000	-0.014245622085722080000000
-0.004818389589117130000000	0.008874581518172193000000
-0.002288706817174152000000	0.004215372263473806000000
0.005178047762882066000000	-0.009537004633076375000000
-0.001950882875482916000000	0.003593165383635231000000
-0.002487335041145060000000	0.004581210463761472000000
0.003207201898765350000000	-0.005907071360873293000000
-0.000408219580908514400000	0.000751864743551140900000
-0.002063426938613331000000	0.003800450203942260000000
0.001711620243707201000000	-0.003152487087170922000000
0.000278188164026471700000	-0.000512371344914807200000
-0.001299494282945251000000	0.002393428678665120000000
0.000619034497683853500000	-0.001140147113165882000000
0.000269179398398805800000	-0.000495778625664020700000

TABLE 4.10. NORMAL MODE AMPLITUDES FOR BIFURCATED WAVEGUIDE

p = 18

% Noise = 0.0

Method - Optimization

Real(x)	Imaginary(x)
-0.5651271771500234000000000	-0.8250038022022767000000000
0.1589366664660428000000000	-0.3015211511355319000000000
0.0284631279897393900000000	-0.0539978301244264300000000
-0.0574064797458812700000000	0.1089067000111651000000000
0.0205920971244315600000000	-0.0390655785555015800000000
0.0176275737885480200000000	-0.0334415365484341400000000
-0.0236441667726119000000000	0.0448557059951115800000000
0.0044091006315241160000000	-0.0083645714197677870000000
0.0132180244199123900000000	-0.0250761138221265900000000
-0.0124499163056562900000000	0.0236189243136035500000000
-0.0010249649592919670000000	0.0019444765092668660000000
0.0104264309694853600000000	-0.0197801397123230300000000
-0.0068763446949482210000000	0.0130452174069262300000000
-0.0034953938248281280000000	0.0066311644325888270000000
0.0085851815532825640000000	-0.0162870776272356000000000
-0.0035811755490244880000000	0.0067939022375628000000000
-0.0051993546230948390000000	0.0098637742056769140000000
0.0079669691969732470000000	-0.0151142576262593700000000
-0.0012932953005069440000000	0.0024535300533226780000000
-0.0095460782573350940000000	0.0181100092808286700000000
0.0164964373320757300000000	-0.0312956404850722100000000
-0.0158733478044230900000000	0.0301135679282827200000000
0.0101404606960294600000000	-0.0192376211843818400000000
-0.0041811699455688240000000	0.0079321606709701670000000
0.0008827680442359446000000	-0.0016747125932062820000000

TABLE 4.11. NORMAL MODE AMPLITUDES FOR BIFURCATED WAVEGUIDE

p = 10

% Noise = 5.0

Method - Optimization

Real(x)	Imaginary(x)
1.34901590398210900000000000	0.14054822201866670000000000
-0.58630267189123020000000000	0.08899796961128060000000000
-0.24997759494626530000000000	0.02160846848204223000000000
0.82130257961543840000000000	-0.04650155321113862000000000
-0.43959502745067040000000000	0.01996564939215598000000000
-0.53561194225611710000000000	0.02036870244671560000000000
1.01136375041092500000000000	-0.03432146132159092000000000
-0.25909915397352010000000000	0.00811208708329217200000000
-1.08160586686641700000000000	0.03051043647999466000000000
1.43125530518927900000000000	-0.03780799385751798000000000
0.16907779395425180000000000	-0.00419490086216371600000000
-2.54175849513747300000000000	0.05969079461349390000000000
2.60524320244879300000000000	-0.05855303991343848000000000
2.22646281930697700000000000	-0.04820185691134818000000000
-10.44858344343887000000000000	0.21957317747240860000000000
10.77579043159127000000000000	-0.22125658878052490000000000
95.11890430550058000000000000	-1.92748874616929600000000000
161.64686117648270000000000000	-3.29769599600417700000000000
104.73971818552670000000000000	-2.33896762403630200000000000
-45.40284199822367000000000000	0.37346391760440820000000000
151.95848829042490000000000000	2.35501025294217000000000000
144.95568702997890000000000000	2.35062982422044500000000000
-77.02610952753533000000000000	1.26353922149146000000000000
-23.21683017645618000000000000	0.38244948326031590000000000
-3.15779388298783700000000000	0.05219856106481736000000000

TABLE 4.12. NORMAL MODE AMPLITUDES FOR BIFURCATED WAVEGUIDE

p = 12

% Noise = 5.0

Method - Optimization

Real(x)	Imaginary(x)
-1.00235852907896300000000	-0.97846941489009990000000
0.31006469663221690000000	-0.39036250194924610000000
0.07640432932115060000000	-0.07326909247967320000000
-0.21888180876895830000000	0.15666031392354940000000
0.10328265822431210000000	-0.05938591566319965000000
0.11044622545673840000000	-0.05271596550886881000000
-0.18938371950060250000000	0.07495844540923930000000
0.04307408267541654000000	-0.01494281063800855000000
0.15919547805807230000000	-0.04755374265641812000000
-0.18915413688262730000000	0.04822028795974246000000
-0.01898409474760631000000	0.00427283011460300100000
0.24906345334621350000000	-0.04854234238360052000000
-0.21012999216720980000000	0.03592147873831490000000
-0.14070347294959210000000	0.02093611497936533000000
0.46126471319032090000000	-0.06069309210420452000000
-0.27065317449151360000000	0.03100929012301441000000
-0.58034860173111150000000	0.05845200987226794000000
1.45982428861318400000000	-0.12819256672973220000000
-0.47091153395508870000000	0.03595925397544772000000
-12.66528153597368000000000	0.83078206101115800000000
-27.01863183532659000000000	1.51520241804752600000000
-29.34806333460518000000000	1.39361160651563600000000
-18.84534511069953000000000	0.74893150443039030000000
-6.94926967056190300000000	0.22808724812290620000000
-1.16247524392765500000000	0.03113081870885036000000

TABLE 4.13. NORMAL MODE AMPLITUDES FOR BIFURCATED WAVEGUIDE

p = 14

% Noise = 5.0

Method - Optimization

Real(x)	Imaginary(x)
-0.5535550281324885000000000	-0.8295446347429494000000000
0.1635719855462542000000000	-0.3038167382678901000000000
0.0311988610014427800000000	-0.0532273252857062000000000
-0.0611417713221645700000000	0.1076639475740755000000000
0.0214448255835784900000000	-0.0392169717302455000000000
0.0228339394367208700000000	-0.0329356018593202500000000
-0.0289312970037428600000000	0.0437287867613229900000000
0.0046510908638282310000000	-0.0080222310723188200000000
0.0152934336815745400000000	-0.0239326092889081100000000
-0.0134289209924681100000000	0.0226464692343510100000000
-0.0011709794127824960000000	0.0019512956885194050000000
0.0115087258743531200000000	-0.0184954540245556200000000
-0.0075709684507100940000000	0.0121338718529567400000000
-0.0033206474884217900000000	0.0059410557979731360000000
0.0090045487049005090000000	-0.0144562914763565400000000
-0.0025603285071671670000000	0.0058493771931077940000000
-0.0055217424978118710000000	0.0081905055584625310000000
0.0093050164375639800000000	-0.0116898921998918000000000
-0.0016335836529978510000000	0.0017496585025476980000000
-0.0085204231626269930000000	0.0105138293487714900000000
0.0100238654750026300000000	-0.0114963364050298100000000
0.0036803951979292240000000	-0.0029215077775925260000000
-0.0390000126188903100000000	0.0352751703289899600000000
-0.0493315492269055200000000	0.0387548632824920100000000
-0.0213338045099557500000000	0.0128830485410871100000000

TABLE 4.14. NORMAL MODE AMPLITUDES FOR BIFURCATED WAVEGUIDE

p = 16

% Noise = 5.0

Method - Optimization

Real(x)	Imaginary(x)
-0.533150533090043700000000	-0.852468416338183500000000
0.159308563250380300000000	-0.299103901033546700000000
0.034076350036204010000000	-0.054220187459914720000000
-0.051150639651232260000000	0.104375139828641000000000
0.018831187632211420000000	-0.036435193276160200000000
0.017815160632070380000000	-0.030201856520201110000000
-0.022875490740330960000000	0.039437638748053540000000
0.001291337792340592000000	-0.007410521109303741000000
0.011605032921866850000000	-0.020779224475529030000000
-0.010795099290414780000000	0.018860031264263250000000
-0.000426200779378132400000	0.001368532271338902000000
0.010170320724947080000000	-0.014370304228535140000000
-0.004764187654901779000000	0.008925454273928310000000
-0.001381241771435778000000	0.004282915809868187000000
0.006746980873245528000000	-0.009625592662225875000000
-0.001651374610394311000000	0.003619194245966769000000
-0.002228188724279366000000	0.004663543116684347000000
0.003481822354194481000000	-0.005971562280503438000000
-0.000935183728774334100000	0.000773321238881363500000
-0.001592075376863637000000	0.003816863333052920000000
0.001145463016529114000000	-0.003182848258040938000000
0.001282928579069627000000	-0.000558644051089136400000
-0.001387939360445013000000	0.002418713344326295000000
0.000128028462173830800000	-0.001155870904897516000000
-0.000260283383492767300000	-0.000499691901020343900000

TABLE 4.15. NORMAL MODE AMPLITUDES FOR BIFURCATED WAVEGUIDE

p = 18

% Noise = 5.0

Method - Optimization

Real(x)	Imaginary(x)
-0.5756673958387165000000000	-0.8446008654762231000000000
0.1528958902455518000000000	-0.3121877427107791000000000
0.0281476301297940200000000	-0.0574780132759763000000000
-0.0538862103246514800000000	0.1113921809626813000000000
0.0203037128513620800000000	-0.0393079969447663000000000
0.0176468163534723300000000	-0.0342918825747588100000000
-0.0264026722677921400000000	0.0455691590856567000000000
0.0064070701861388890000000	-0.0083140685090928410000000
0.0147760271128326600000000	-0.0256588566117524200000000
-0.0122956545024289200000000	0.0241537726341783200000000
-0.0005083578254922163000000	0.0019421881719146580000000
0.0114715450530849300000000	-0.0201286945337159600000000
-0.0091135931936450570000000	0.0132990588935517800000000
-0.0041377686720306980000000	0.0066388490766845080000000
0.0084795681701519340000000	-0.0165578391221838600000000
-0.0045609385308836570000000	0.0068705984302362800000000
-0.0062437489536120490000000	0.0100731914298415500000000
0.0104395129488849200000000	-0.0154539361167026100000000
-0.0025787959228411210000000	0.0025173604248903590000000
-0.0124019304870910400000000	0.0185906849200285900000000
0.0222093127545249100000000	-0.0321064875567949100000000
-0.0222559640900727100000000	0.0309185110096987100000000
0.0153254116355631300000000	-0.0197908730864107000000000
-0.0078002730530776860000000	0.0081467347535768300000000
0.0026957561223150330000000	-0.0017230291172084940000000

5. SECOND EXAMPLE

In the next example, an attempt was made to solve an integral equation related to an electromagnetic remote sensing problem.[4] The equation is

$$\int_{o}^{1} f(x) \quad \sinh^2 y(x - 1) \, dx = g(y) \quad . \tag{5}$$

The method of moments was used to convert this continuous operator equation to a matrix equation.

Let f(x) be approximated by a sum of expansion functions, so that

$$f(x) \simeq \sum_{j=1}^{N} \alpha_j \phi_j (x) \tag{6}$$

where $\phi_j(x)$ is an expansion function and α_j is its respective coefficient. We note that the operator of Equation (5) is linear which permits us to distribute the integration over the series approximating f(x). Thus,

$$\sum_{j=1}^{N} \alpha_j \int_{o}^{1} \phi_j (x) \quad \sinh^2 y(x - 1) \, dx = g(y) \ .$$

We next take the inner product of both sides with the testing functions $\delta(y - y_i)$ which results in

$$\sum_{j=1}^{N} \alpha_j \int_{o}^{1} \phi_j (x) \ \sinh^2 y_i (x - 1) \, dx = g(y_i) \quad . \tag{7}$$

To finally get this into matrix form, we define

$$A_{ij} = \int_0^1 \phi_j(x) \sinh^2 y_i(x - 1)\, dx$$

$$g(y_i) = \beta_i \quad .$$

Equation (7) can now be written as

$$\sum_{j=1}^{E} \alpha_j A_{ij} = \beta_i \quad .$$

Expressed in matrix form, we get

$$\begin{bmatrix} \\ A \\ \\ \end{bmatrix} \begin{bmatrix} \\ \alpha \\ \\ \end{bmatrix} = \begin{bmatrix} \\ \beta \\ \\ \end{bmatrix} \quad .$$

To effect a solution on the computer, we must specify the expansion functions $\phi_j(x)$. In this paper,

$$\phi_j(x) = x^{j-1} \quad .$$

A_{ij} is then found to be

$$A_{ij} = \int_0^1 x^{j-1} \sinh^2 y_i(x - 1)\, dx \quad .$$

To carry out the integration, we make the substitutions

$$u = y_i(x - 1)$$

$$du = y_i\, dx$$

$$A_{ij} = \frac{1}{y_i} \int_{-y_i}^{0} \left(\frac{u}{y_i} + 1\right)^{j-1} \sinh^2 u\, du \quad .$$

Using the binomial expansion theorem,

$$A_{ij} = \int_{-y_i}^{o} \frac{1}{y_i}\left[\sum_{k=o}^{j-1} \frac{(j-1)!}{(j-k-1)!k!}\left(\frac{u}{y_i}\right)^{k}\right] \sinh^2 u \, du$$

$$= \int_{-y_i}^{o} \frac{1}{y_i}\left[\sum_{k=1}^{j} \frac{(j-1)!}{(j-k)!(k-1)!}\left(\frac{u}{y_i}\right)^{k-1}\right] \sinh^2 u \, du$$

$$= \sum_{k=1}^{j} y_i^{-k} \frac{(j-1)!}{(j-k)!(k-1)!} S(k)$$

where

$$S(k) = \int_{-y_i}^{o} x^{k-1} \sinh^2 x \, dx \quad .$$

From an integral table[5] we find

$$\int x^n \sinh^2 (x) \, dx = \frac{-x^{n+1}}{2(n+1)} + \frac{n!}{4} \sum_{k=o}^{E\left(\frac{n}{2}\right)} \left[\frac{x^{n-2k} \sinh (2x)}{2^{2k} (n-2k)!} - \frac{x^{n-2k-1} \cosh(2x)}{2^{2k+1}(n-2k-1)!}\right].$$

We now have an explicit formula for the A matrix.

Before going further, we must choose the domain of g(y), the number of expansion functions, and the number of testing functions. Although it is possible to have more testing functions than expansion functions, it is simpler to generate the solution if there are an equal number of the two types of functions and, thus, a square matrix. In this section, solutions were sought for varying numbers of expansion functions and for varying domains of g(y).

The testing procedure was as follows. An $f(x)$ was assumed from which the right-hand side of Equation (5) could be calculated exactly. The inner product of this $g(y)$ was then taken with the testing functions $\delta(y - y_i)$ to find the elements of the matrix. Then it was attempted to reconstruct the assumed $f(x)$ given the A and β matrices. A solution was attempted for each of four assumed functions: $f(x) = 1$, $f(x) = x^2$, $f(x) = \sin(\pi x)$, $f(x) = \sin(5\pi x)$.

The coefficients were found by using both Gaussian elimination and optimization techniques. The Gaussian elimination method resulted in the more exact solution. The same problems of overflow were again encountered when using DFMFP for larger domains of $g(y)$. When the function value and gradient were multiplied by a number within FUNCT, answers were obtainable for larger domains of $g(y)$, but they were much less accurate. The answers reported were found without using any multiplying factor.

After the coefficients for the expansion functions were found, the power series approximation to $f(x)$ was evaluated at various values of x. These values along with the true values are tabulated in the following tables.

TABLE 5.1. RECONSTRUCTED FUNCTION VALUES

Domain of g(y) is {y: 0.0 < y < YE}

N = 4

Method - Optimization

YE= 2.50

F(X)=1

CORRECT	CALCULATED	ERROR
1.00000000000	1.00032439684	-0.324396840952D-03
1.00000000000	0.999317671743	0.682328256995D-03
1.00000000000	1.00011673488	-0.116734879538D-03
1.00000000000	1.00894447917	-0.894447916742D-02

F(X)=SIN(PI*X)

CORRECT	CALCULATED	ERROR
0.382683359860	0.343217595064	0.394657647956D-01
0.923879442413	0.749150436478	0.174729005936
0.923879682675	1.33387338504	-0.409993702365
0.382683939902	2.15369604060	-1.77101210070

F(X)=X**2

CORRECT	CALCULATED	ERROR
0.156250000000D-01	0.205291063448D-01	-0.490410634477D-02
0.140625000000	0.143298227823	-0.267322782335D-02
0.390625000000	0.372290024440	0.183349755601D-01
0.765625000000	0.747747362151	0.178776378488D-01

F(X)=SIN(5*PI*X)

CORRECT	CALCULATED	ERROR
0.923879682675	0.395055487014	0.528824195661
-0.382684519944	-0.181708889698D-01	-0.364513630974
-0.382681619733	-0.486904539040	0.104222919307
0.923878481365	-1.01009604106	1.93397452243

TABLE 5.2. RECONSTRUCTED FUNCTION VALUES

Domain of g(y) is {y: 0.0 < y < YE}

N = 4

Method - Optimization

YE= 5.00

F(X)=1

CORRECT	CALCULATED	ERROR
1.00000000000	1.00036753360	-0.367533598703D-03
1.00000000000	0.999159300662	0.840699338224D-03
1.00000000000	1.00044495943	-0.444959430271D-03
1.00000000000	1.01203072637	-0.120307263687D-01

F(X)=SIN(PI*X)

CORRECT	CALCULATED	ERROR
0.382683359860	0.386640347517	-0.395698765678D-02
0.923879442413	0.916829181261	0.705026115271D-02
0.923879682675	0.946378140612	-0.224984579378D-01
0.382683939902	0.286561483641	0.961224562607D-01

F(X)=X**2

CORRECT	CALCULATED	ERROR
0.156250000000D-01	0.181273364432D-01	-0.250233644318D-02
0.140625000000	0.134899954776	0.572504522359D-02
0.390625000000	0.393660842679	-0.303584267861D-02
0.765625000000	0.847570483165	-0.819454831652D-01

F(X)=SIN(5*PI*X)

CORRECT	CALCULATED	ERROR
0.923879682675	0.383414934051	0.540464748623
-0.382684519944	-0.112563400807	-0.270121119137
-0.382681619733	-0.380118460258	-0.256315947507D-02
0.923878481365	-0.314911510689	1.23878999205

TABLE 5.3. RECONSTRUCTED FUNCTION VALUES

Domain of g(y) is {y: 0.0 < y < YE}

N = 4

Method - Optimization

YE= 10.0

F(X)=1

CORRECT	CALCULATED	ERROR
1.00000000000	1.00014282939	-0.142829389366D-03
1.00000000000	0.999126782873	0.873217127032D-03
1.00000000000	1.00849210828	-0.849210827702D-02
1.00000000000	1.04374297373	-0.437429737327D-01

F(X)=SIN(PI*X)

CORRECT	CALCULATED	ERROR
0.382683359860	0.384247884274	-0.156452441388D-02
0.923879442413	0.917885451900	0.599399051295D-02
0.923879682675	1.02530110059	-0.101421417919
0.382683939902	0.586273513597	-0.203589573695

F(X)=X**2

CORRECT	CALCULATED	ERROR
0.156250000000D-01	0.161546194444D-01	-0.529619444390D-03
0.140625000000	0.137374034664	0.325096533635D-02
0.390625000000	0.421627555424	-0.310025554243D-01
0.765625000000	0.925855142525	-0.160230142525

F(X)=SIN(5*PI*X)

CORRECT	CALCULATED	ERROR
0.923879682675	0.699458232406	0.224421450269
-0.382684519944	-0.593549636234	0.210865116290
-0.382681619733	-5.43017826864	5.04749664890
0.923878481365	-14.7377692041	15.6616476855

TABLE 5.4. RECONSTRUCTED FUNCTION VALUES

Domain of g(y) is {y: 0.0 < y < YE}

N = 8

Method - Optimization

YE= 2.50

F(X)=1

CORRECT	CALCULATED	ERROR
1.00000000000	1.00393485659	-0.393485658955D-02
1.00000000000	0.997453871562	0.254612843779D-02
1.00000000000	0.988221451970	0.117785480304D-01
1.00000000000	0.981530657663	0.184693423374D-01
1.00000000000	0.990776725528	0.922327447190D-02
1.00000000000	1.04472094405	-0.447209440476D-01
1.00000000000	1.19851648324	-0.198516483237
1.00000000000	1.54933565809	-0.549335658094

F(X)=SIN(PI*X)

CORRECT	CALCULATED	ERROR
0.195090283531	0.273422296279	-0.783320127484D-01
0.555570135140	0.420492266405	0.135077868735
0.831469503301	0.602170243084	0.229299260217
0.980785226816	0.837269729007	0.143515497809
0.980785349300	1.15972775755	-0.178942408249
0.831469852106	1.62978560560	-0.798315753496
0.555570657164	2.35054786118	-1.79497720402
0.195090899300	3.49107944747	-3.29598854817

F(X)=X**2

CORRECT	CALCULATED	ERROR
0.390625000000D-02	0.114546027231D-01	-0.754835272309D-02
0.351562500000D-01	0.337968323559D-01	0.135941764415D-02
0.976562500000D-01	0.755447610738D-01	0.221114889262D-01
0.191406250000	0.152341162766	0.390650872341D-01
0.316406250000	0.293917244040	0.224890059595D-01
0.472656250000	0.554855542663	-0.821992926629D-01
0.660156250000	1.03058816351	-0.370431913514
0.878906250000	1.87976442315	-1.00085817315

F(X)=SIN(5*PI*X)

CORRECT	CALCULATED	ERROR
0.831469503301	0.505532915832	0.325936587469
0.195090899300	0.275134140227	-0.800432409267D-01
-0.980785471784	0.425798846048D-01	-1.02336535639
0.555569091092	-0.177764949109	0.733334040200
0.555571701211	-0.356365250965	0.911936952176
-0.980784859362	-0.436689979391	-0.544094879970
0.195087820453	-0.317627525943	0.512715346396
0.831471247324	0.171103683967	0.660367563357

TABLE 5.5. RECONSTRUCTED FUNCTION VALUES

Domain of g(y) is {y: 0.0 < y < YE}

N = 8

Method – Optimization

YE= 5.00

F(X)=1

CORRECT	CALCULATED	ERROR
1.00000000000	1.00072335365	-0.723353645402D-03
1.00000000000	1.00285905390	-0.285905389923D-02
1.00000000000	0.996954056567	0.304594343348D-02
1.00000000000	0.987057559989	0.129424400106D-01
1.00000000000	0.985204545105	0.147954548953D-01
1.00000000000	1.01880960839	-0.188096083902D-01
1.00000000000	1.14191829994	-0.141918299943
1.00000000000	1.45117868134	-0.451178681338

F(X)=SIN(PI*X)

CORRECT	CALCULATED	ERROR
0.195090283531	0.197705476216	-0.261519268551D-02
0.555570135140	0.554010517714	0.155961742567D-02
0.831469503301	0.825475048747	0.599445455364D-02
0.980785226816	0.985402049239	-0.461682242309D-02
0.980785349300	1.00242877598	-0.216434266774D-01
0.831469852106	0.842342751456	-0.108728993498D-01
0.555570657164	0.471787692284	0.837829648798D-01
0.195090899300	-0.135632573267	0.330723472567

F(X)=X**2

CORRECT	CALCULATED	ERROR
0.390625000000D-02	0.587914934171D-02	-0.197289934171D-02
0.351562500000D-01	0.407165070652D-01	-0.556025706516D-02
0.976562500000D-01	0.898727844427D-01	0.778346555733D-02
0.191406250000	0.165785868885	0.256203811150D-01
0.316406250000	0.292966269075	0.234399809247D-01
0.472656250000	0.517547572020	-0.448913220200D-01
0.660156250000	0.921556305699	-0.261400055699
0.878906250000	1.64293166159	-0.764025411595

F(X)=SIN(5*PI*X)

CORRECT	CALCULATED	ERROR
0.831469503301	0.540639791880	0.290829711421
0.195090899300	0.309035825687	-0.113944926387
-0.980785471784	-0.292765693744D-01	-0.951508902410
0.555569091092	-0.396834904775	0.952403995866
0.555571701211	-0.622275694373	1.17784739558
-0.980784859362	-0.369881297206	-0.610903562156
0.195087820453	0.963790692757	-0.768702872305
0.831471247324	4.39311430002	-3.56164305270

TABLE 5.6. RECONSTRUCTED FUNCTION VALUES

Domain of g(y) is {y: 0.0 < y < YE}

N = 8

Method - Optimization

YE= 10.0

F(X)=1

CORRECT	CALCULATED	ERROR
1.00000000000	1.00005886241	-0.588624053408D-04
1.00000000000	0.999973701066	0.262989335540D-04
1.00000000000	0.998955318116	0.104468188377D-02
1.00000000000	0.998873252508	0.112674749162D-02
1.00000000000	1.00798604263	-0.798604263415D-02
1.00000000000	1.04755181393	-0.475518139313D-01
1.00000000000	1.16201968226	-0.162019682260
1.00000000000	1.43361575539	-0.433615755386

F(X)=SIN(PI*X)

CORRECT	CALCULATED	ERROR
0.195090283531	0.196138940103	-0.104865657244D-02
0.555570135140	0.553063976003	0.250615913690D-02
0.831469503301	0.833753063662	-0.228356036079D-02
0.980785226816	0.998249004789	-0.174637779723D-01
0.980785349300	0.991528839861	-0.107434905603D-01
0.831469852106	0.738916325516	0.925535265897D-01
0.555570657164	0.140665754719	0.414904902445
0.195090899300	-0.934318762295	1.12940966159

F(X)=X**2

CORRECT	CALCULATED	ERROR
0.390625000000D-02	0.386931805631D-02	0.369319436887D-04
0.351562500000D-01	0.352509547590D-01	-0.947047590475D-04
0.976562500000D-01	0.976878746800D-01	-0.316246800224D-04
0.191406250000	0.190475302882	0.930947117871D-03
0.316406250000	0.314587700686	0.181854931389D-02
0.472656250000	0.476087901865	-0.343165186505D-02
0.660156250000	0.691732792335	-0.315765423354D-01
0.878906250000	0.997311709271	-0.118405459271

F(X)=SIN(5*PI*X)

CORRECT	CALCULATED	ERROR
0.831469503301	0.843253556418	-0.117840531169D-01
0.195090899300	0.276768128120	-0.816772288200D-01
-0.980785471784	-0.996401167874	0.156156960898D-01
0.555569091092	-0.417389864164	0.972958955255
0.555571701211	2.55103363604	-1.99546193483
-0.980784859362	3.68202472197	-4.66280958134
0.195087820453	-10.2533522909	10.4484401114
0.831471247324	-67.5532731736	68.3847444209

TABLE 5.7. RECONSTRUCTED FUNCTION VALUES

Domain of g(y) is {y: 0.0 < y < YE}

N = 4

Method - Gauss

YE = 2.5

F(X)=1

CORRECT	CALCULATED	ERROR
1.00000000000	1.00000000000	0.444699832514D-12
1.00000000000	1.00000000000	-0.591304782915D-12
1.00000000000	1.00000000000	-0.352384788016D-12
1.00000000000	0.999999999992	0.830882584957D-11

F(X)=SIN(P*X)

CORRECT	CALCULATED	ERROR
0.382683359860	0.386530753386	-0.384739352603D-02
0.923879442413	0.918154180182	0.572526223089D-02
0.923879682675	0.943836456584	-0.199567739099D-01
0.382683939902	0.266374767586	0.116309172316

F(X)=X**2

CORRECT	CALCULATED	ERROR
0.156250000000D-01	0.156249999996D-01	0.429978101735D-12
0.140625000000	0.140625000001	-0.570279934387D-12
0.390625000000	0.390625000000	-0.344529960117D-12
0.765625000000	0.765624999992	0.800945421098D-11

F(X)=SIN(5*PI*X)

CORRECT	CALCULATED	ERROR
0.923879682675	0.412510282984	0.511369399691
-0.382684519944	-0.273710561743	-0.108973958200
-0.382681619733	-0.132218047460	-0.250463572273
0.923878481365	2.10208631203	-1.17820783067

TABLE 5.8. RECONSTRUCTION FUNCTION VALUES

Domain of g(y) is {y: 0.0 < y < YE}

N = 4

Method - Gauss

YE = 5.0

F(X)=1

CORRECT	CALCULATED	ERROR
1.00000000000	1.00000000000	0.525968157916D-14
1.00000000000	1.00000000000	-0.910382880193D-14
1.00000000000	1.00000000000	-0.155431223448D-14
1.00000000000	1.00000000000	0.119002030452D-12

F(X)=SIN(PI*X)

CORRECT	CALCULATED	ERROR
0.382683359860	0.385155669915	-0.247321005537D-02
0.923879442413	0.920226522169	0.365292024469D-02
0.923879682675	0.944574260642	-0.206945779674D-01
0.382683939902	0.237926744807	0.144757195095

F(X)=**2

CORRECT	CALCULATED	ERROR
0.156250000000D-01	0.156250000000D-01	0.951495826573D-15
0.140625000000	0.140625000000	0.159594559790D-14
0.390625000000	0.390625000000	-0.661970478433D-14
0.765625000000	0.765625000000	-0.270339306496D-13

F(X)=SIN(5*PI*X)

CORRECT	CALCULATED	ERROR
0.923879682675	0.534684591614	0.389195091061
-0.382684519944	-0.458650735253	0.759662153098D-01
-0.382681619733	-0.196595332038	-0.186086287695
0.923878481365	4.63881812009	-3.71493963872

TABLE 5.9. RECONSTRUCTED FUNCTION VALUES

Domain of g(y) is {y: 0.0 < y < YE}

N = 4

Method - Gauss

YE = 10.0

F(X) = 1

CORRECT	CALCULATED	ERROR
1.00000000000	1.00000000000	0.582867087928D-15
1.00000000000	1.00000000000	-0.177635683940D-14
1.00000000000	1.00000000000	0.459354776439D-14
1.00000000000	1.00000000000	0.367206265395D-13

F(X)=SIN(PI*X)

CORRECT	CALCULATED	ERROR
0.382683359860	0.382767600379	-0.842405191514D-04
0.923879442413	0.926951138269	-0.307169585562D-02
0.923879682675	0.937890041604	-0.140103589297D-01
0.382683939902	0.135359441899	0.247324498003

F(X)=X**2

CORRECT	CALCULATED	ERROR
0.156250000000D-01	0.156250000000D-01	-0.160548657702D-14
0.140625000000	0.140625000000	0.846545056277D-15
0.390625000000	0.390625000000	0.945077349712D-14
0.765625000000	0.765625000000	0.141692213518D-13

F(X)=SIN(5*PI*X)

CORRECT	CALCULATED	ERROR
0.923879682675	0.801707075221	0.122172607453
-0.382684519944	-1.22038659825	0.837702078308
-0.382681619733	0.579689758030	-0.962371377763
0.923878481365	16.2945681660	-15.3706896846

TABLE 5.10. RECONSTRUCTED FUNCTION VALUES

Domain of g(y) is {y: 0.0 < y < YE}

N = 8

Method - Gauss

YE = 2.5

F(X)=1

CORRECT	CALCULATED	ERROR
1.00000000000	1.00000013027	-0.130265724207D-06
1.00000000000	0.999999857246	0.142754472132D-06
1.00000000000	1.00000024124	-0.241237670728D-06
1.00000000000	0.999999892182	0.107818383319D-06
1.00000000000	0.999999411834	0.588165838311D-06
1.00000000000	1.00000110476	-0.110476367632D-05
1.00000000000	1.00000158535	-0.158534871408D-05
1.00000000000	0.999978527908	0.214720919098D-04

F(X)=SIN(PI*X)

CORRECT	CALCULATED	ERROR
0.195090283531	0.195085993822	0.428970823621D-05
0.555570135140	0.555577915809	-0.778066926425D-05
0.831469503301	0.831459612337	0.989096349792D-05
0.980785226816	0.980779141549	0.608526767940D-05
0.980785349300	0.980836579725	-0.512304246149D-04
0.831469852106	0.831411302485	0.585496207740D-04
0.555570657164	0.555367676864	0.202980300447D-03
0.195090899300	0.197091066722	-0.200016742257D-02

F(X)=**2

CORRECT	CALCULATED	ERROR
0.390625000000D-02	0.390586223766D-02	0.387762337962D-06
0.351562500000D-01	0.351566739390D-01	-0.423938958122D-06
0.976562500000D-01	0.976555334236D-01	0.716576379609D-06
0.191406250000	0.191406572877	-0.322876702860D-06
0.316406250000	0.316407986865	-0.173686500167D-05
0.472656250000	0.472652977554	0.327244558852D-05
0.660156250000	0.660151602768	0.464723234163D-05
0.878906250000	0.878969452133	-0.632021333217D-04

F(X)=SIN(5*PI*X)

CORRECT	CALCULATED	ERROR
0.831469503301	0.898234618366	-0.667651150647D-01
0.195090899300	0.149974476070	0.451164232299D-01
-0.980785471784	-0.839473802652	-0.141311669133
0.555569091092	0.173525822960	0.382043268131
0.555571701211	1.14585503368	-0.590283332468
-0.980784859362	-1.06724381390	0.864589545368D-01
0.195087820453	-2.54522335865	2.74031117911
0.831471247324	19.8919035871	-19.0604323397

TABLE 5.11. RECONSTRUCTED FUNCTION

Domain of g(y) is {y: 0.0 < y < YE}

N = 8

Method - Gauss

YE = 5.0

F(X)=1

CORRECT	CALCULATED	ERROR
1.00000000000	0.999999999284	0.715634690396D-09
1.00000000000	0.999999999250	0.750485854017D-09
1.00000000000	0.999999999488	0.512244108242D-09
1.00000000000	1.00000000650	-0.650166898097D-08
1.00000000000	0.999999984727	0.152729624225D-07
1.00000000000	1.00000000406	-0.406042865997D-08
1.00000000000	1.00000009559	-0.955863623808D-07
1.00000000000	0.999999241948	0.758052063662D-06

F(X)=SIN(PI*X)

CORRECT	CALCULATED	ERROR
0.195090283531	0.195089881285	0.402245829756D-06
0.555570135140	0.555570264242	-0.129101766139D-06
0.831469503301	0.831468443880	0.105942092377D-05
0.980785226816	0.980789305854	-0.407903726130D-05
0.980785349300	0.980778477412	0.687188812204D-05
0.831469852106	0.831464966588	0.488551805439D-05
0.555570657164	0.555644633901	-0.739767369056D-04
0.195090899300	0.194541305072	0.549594228093D-03

F(X)=X**2

CORRECT	CALCULATED	ERROR
0.390625000000D-02	0.390625207842D-02	-0.207841641554D-08
0.351562500000D-01	0.351562513982D-01	-0.139816589903D-08
0.976562500000D-01	0.976562521339D-01	-0.213385389913D-08
0.191406250000	0.191406233669	0.163310080875D-07
0.316406250000	0.316406284462	-0.344617861608D-07
0.472656250000	0.472656245736	0.426397697706D-08
0.660156250000	0.660156026616	0.223383635645D-06
0.878906250000	0.878907968205	-0.171820503503D-05

F(X)=SIN(5*PIX)

CORRECT	CALCULATED	ERROR
0.831469503301	0.832577376712	-0.110787341147D-02
0.195090899300	0.148003583170	0.470873161298D-01
-0.980785471784	-0.927927000837	-0.528584709468D-01
0.555569091092	0.524168699037	0.314003920541D-01
0.555571701211	0.562384312986	-0.681261177493D-02
-0.980784859362	-1.19389945458	0.213114595217
0.195087820453	1.65577018395	-1.46068236350
0.831471247324	-10.6135982335	11.4450694808

TABLE 5.12. RECONSTRUCTED FUNCTION VALUES

Domain of g(y) is {y: 0.0 < y < YE}

N = 8

Method - Gauss

YE = 10.0

F(X)=1

CORRECT	CALCULATED	ERROR
1.00000000000	1.00000000000	-0.945910016981D-13
1.00000000000	1.00000000000	-0.496247487547D-11
1.00000000000	0.999999999994	0.585063941738D-11
1.00000000000	0.999999999988	0.124843885230D-10
1.00000000000	1.00000000007	-0.690389967417D-10
1.00000000000	0.999999999925	0.750776524283D-10
1.00000000000	0.999999999631	0.368919728100D-09
1.00000000000	1.00000000363	-0.362715968372D-08

F(X)=SIN(P*X)

CORRECT	CALCULATED	ERROR
0.195090283531	0.195090186347	0.971837593733D-07
0.555570135140	0.555570455895	-0.320754682342D-06
0.831469503301	0.831468790285	0.713015537293D-06
0.980785226816	0.980786902937	-0.167612106555D-05
0.980785349300	0.980783300617	0.204868387196D-05
0.831469852106	0.831464802191	0.504991541872D-05
0.555570657164	0.555613007649	-0.423504850297D-04
0.195090899300	0.194778883599	0.312015700421D-03

F(X)=X**2

CORRECT	CALCULATED	ERROR
0.390625000000D-02	0.390625000018D-02	-0.176854191014D-12
0.351562500000D-01	0.351562499950D-01	0.496907202885D-11
0.976562500000D-01	0.976562500071D-01	-0.713074044256D-11
0.191406250000	0.191406250009	-0.866283433876D-11
0.316406250000	0.316406249930	0.698968799506D-10
0.472656250000	0.472656250094	-0.939552047274D-10
0.660156250000	0.660156250365	-0.364804006447D-09
0.878906250000	0.878906246126	0.387429102688D-08

F(X)=SIN(5*PI*X)

CORRECT	CALCULATED	ERROR
0.831469503301	0.844347164257	-0.128776609562D-01
0.195090899300	0.164836718532	0.302541807675D-01
-0.980785471784	-0.924952706385	-0.558327653991D-01
0.555569091092	0.406505386032	0.149063705060
0.555571701211	0.875356958384	-0.319785257172
-0.980784859362	-1.32955040482	0.348765545461
0.195087820453	-0.244829169278	0.439916989730
0.831471247324	5.11032578941	-4.27885454209

6. CONCLUSION

An evaluation of the previous tests must take into account these three areas of performance: 1) accuracy of solution, 2) cost and 3) ease of programming. To make a judgment in the first category we must consider Sections 3 and 5. Only in these sections were the true solutions known. Table 3.1 shows that for the type of matrix in question neither method of solution has a decided advantage over the other. Both have almost the same limiting values for noise levels and condition numbers. For the matrix in Section 5, it was found that Gaussian elimination worked better than the optimization method in every case.

It was found that for identical problems the optimization method cost more than three times the other method. Not only is the cost prohibitive, but as was mentioned earlier, DFMFP had a tendency to go into underflow or overflow. This could be avoided if a multiplying factor was introduced in the subroutine FUNCT. However, it was not clear what this factor should be and thus necessitated trial-and-error. Of course, this is not only inconvenient to the programmer but also adds to the already high cost of the optimization method. It can now be said that in the solution of a matrix equation, this optimization technique holds no advantage over the usual Gaussian elimination and, in fact, is inferior in all categories considered.

LIST OF REFERENCES

1. M. Aoki, Introduction to Optimization Techniques. New York: The Macmillan Company, 1971, pp. 138-146.

2. R. Mittra and S. W. Lee, Analytical Techniques in the Theory of Guided Waves. New York: The Macmillan Company, 1971, pp. 30-37.

3. C. A. Klein, private communications.

4. R. Mittra, D. H. Schaubert and M. Mostafavi, "Some methods for determining the profile functions of inhomogeneous media," NASA Technical Memorandum X-62, 150, Mathematics of Profile Inversion, August 1972, pp. 8.2-8.12.

5. I. S. Gradschteyn and I. M. Ryzhik, Table of Integrals, Series and Products. New York: Academic Press, 1965, p. 121.

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D		
(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)		
1. ORIGINATING ACTIVITY (Corporate author) Department of Electrical Engineering University of Illinois Urbana, Illinois 61801	2a. REPORT SECURITY CLASSIFICATION	2b. GROUP
3. REPORT TITLE A MATRIX EQUATION SOLUTION BY AN OPTIMIZATION TECHNIQUE		
4. DESCRIPTIVE NOTES (Type of report and inclusive dates) Scientific-December 1972		
5. AUTHOR(S) (First name, middle initial, last name) Martin Johnson Raj Mittra		
6. REPORT DATE December 1972	7a. TOTAL NO. OF PAGES 48	7b. NO. OF REFS 5
8a. CONTRACT OR GRANT NO NGR-14-005-009 b. PROJECT NO. c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) Antenna Laboratory Report No. 72-11 Technical Report No. 20 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) UILU-ENG-72-2557	
10. DISTRIBUTION STATEMENT Unlimited		
11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY National Aeronautics and Space Administration	

13. ABSTRACT

The computer solution of matrix equations is often difficult to accomplish due to an ill-conditioned matrix or high noise levels. This paper compares two methods of solution for matrices of various degrees of ill-conditioning and for various noise levels in the right-hand side vector. One method employs the usual Gaussian elimination. The other solves the equation by an optimization technique and employs a function minimization subroutine.

DD FORM 1 NOV 65 1473

UNCLASSIFIED

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Matrix						
Optimization						
Noise						
Ill-Conditioned						