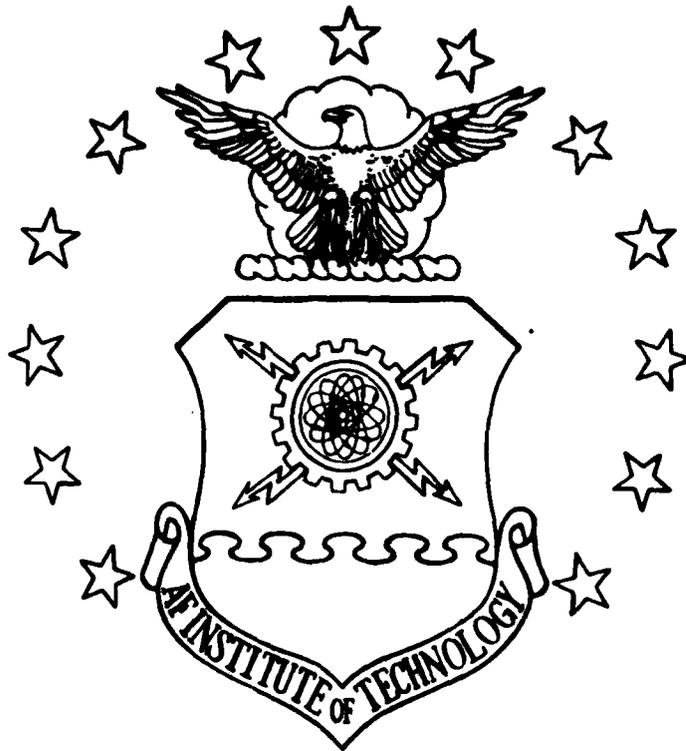


MICROCOPY RESOLUTION TEST CHART  
NATIONAL BUREAU OF STANDARDS-1963-A

AD-A154 434



EVALUATION OF THE EIGENVALUE  
METHOD IN THE SOLUTION OF  
TRANSIENT HEAT CONDUCTION PROBLEMS

THESIS

David W. Landry  
Captain, USAF

AFIT/GNE/ENP/85M-12

|        |
|--------|
| Access |
| NTIS   |
| DTIC   |
| Unann. |
| Justi. |

DTIC FILE COPY

This document has been approved  
for public release and under its  
distribution is unlimited.

DTIC  
ELECTE  
JUN 4 1985  
S E D

DEPARTMENT OF THE AIR FORCE  
AIR UNIVERSITY

**AIR FORCE INSTITUTE OF TECHNOLOGY**

Wright-Patterson Air Force Base, Ohio

85 5 07 177

AFIT/GNE/ENP/85M-12

①

**DTIC ELECTED**  
JUN 4 1985  
**S D E**

**EVALUATION OF THE EIGENVALUE  
METHOD IN THE SOLUTION OF  
TRANSIENT HEAT CONDUCTION PROBLEMS**

**THESIS**

**David W. Landry  
Captain, USAF**

**AFIT/GNE/ENP/85M-12**

|                           |                                     |
|---------------------------|-------------------------------------|
| <b>Accession For</b>      |                                     |
| NTIS GRA&I                | <input checked="" type="checkbox"/> |
| DTIC TAB                  | <input type="checkbox"/>            |
| Unannounced               | <input type="checkbox"/>            |
| Justification             |                                     |
| By _____                  |                                     |
| Distribution/             |                                     |
| <b>Availability Codes</b> |                                     |
| Dist                      | Avail and/or<br>Special             |
| A/1                       |                                     |

Approved for public release; distribution unlimited

DTIC  
COPY  
INSTRUMENTS  
3

AFIT/GNE/ENP/85M-12

EVALUATION OF THE EIGNVALUE METHOD IN  
THE SOLUTION OF TRANSIENT HEAT CONDUCTION PROBLEMS

THESIS

Presented to the Faculty of the School of Engineering  
of the Air Force Institute of Technology  
Air University  
In Partial Fulfillment of the  
Requirements for the Degree of  
Master of Science in Nuclear Engineering

David W. Landry, B.S.

Captain, USAF

January 1985

Approved for public release; distribution unlimited

## Table of Contents

|   | Page |
|---|------|
| Preface . . . . .                                   | iii  |
| List of Figures . . . . .                           | iv   |
| List of Tables . . . . .                            | v    |
| List of Symbols . . . . .                           | vi   |
| Abstract . . . . .                                  | viii |
| I. Introduction . . . . .                           | 1    |
| Problem . . . . .                                   | 2    |
| Scope . . . . .                                     | 2    |
| General Approach . . . . .                          | 3    |
| II. Existing Theory . . . . .                       | 5    |
| Explicit . . . . .                                  | 5    |
| Implicit . . . . .                                  | 7    |
| Crank-Nicolson . . . . .                            | 9    |
| III. New Theory . . . . .                           | 10   |
| What is the EM? . . . . .                           | 10   |
| Why does the EM work? . . . . .                     | 11   |
| Derivation of the method . . . . .                  | 12   |
| Advantages and disadvantages . . . . .              | 15   |
| IV. Application to Specific Problems . . . . .      | 17   |
| The test problem . . . . .                          | 17   |
| Recursion formulae . . . . .                        | 22   |
| Error evaluations . . . . .                         | 26   |
| Results of 2-D problem . . . . .                    | 27   |
| Time comparisons . . . . .                          | 29   |
| Accuracy comparisons . . . . .                      | 31   |
| Extension of sample<br>problem into 3-D . . . . .   | 33   |
| The three dimension<br>problem statement . . . . .  | 33   |
| Three dimensional<br>time results . . . . .         | 36   |
| Accuracy of 3-D results . . . . .                   | 41   |
| Non-symmetric 3-D problem . . . . .                 | 42   |
| V. Evaluation of the<br>Eigenvalue Method . . . . . | 44   |

|   | Page |
|---|------|
| The strong points . . . . .   | 44   |
| The weak points . . . . .   | 46   |
| VI. Conclusions and Recommendations . . . . .   | 48   |
| Conclusions . . . . .   | 48   |
| Recommendations . . . . .   | 49   |
| Appendix A: Derivation of the<br>Analytical Solution . . . . .                                  | 51   |
| Appendix B: Application of the<br>Eigenvalue Method to a<br>Small System of Equations . . . . . | 60   |
| Appendix C: Standard IMSL Routines<br>Used in the Study . . . . .                               | 63   |
| Bibliography . . . . .  | 74   |
| Vita . . . . .  | 76   |

## Preface

The purpose of this thesis was to bring to the surface a new tool in the neverending quest to solve heat conduction problems analytically. This work expands the knowledge of the eigenvalue method a few more steps. I hope this report inspires even further study in this worthy procedure.

I would like to thank my advisor, Dr. B. Kaplan of the Air Force Institute of Technology at Wright-Patterson AFB, Ohio, who had the wisdom to give enough rein in the project to allow myself to conduct the investigation as I envisioned, and yet, held enough rein to insure I completed a quality, well scoped report.

Special thanks to Dr. N. Pagano of AFWAL/MLBM who sponsored this thesis and Dr. J. Jones who made me feel as an equal and as one of the numerical analysis community's citizens.

Finally, I feel deep gratitude and appreciation for my wife, Naomi, who patiently surrendered her husband to a computer for six months.

## List of Figures

| Figure  | Page |
|---|------|
| 1. Notation convention for discretizations . . . . .          | 12   |
| 2. Domain for 2-D transient heat conduction problem . . . . . | 18   |
| 3. History of temperature at nodal point $T_n$ . . . . .      | 24   |
| 4. Accuracy comparison of methods in full grid . . . . .      | 32   |
| 5. CPU time required to run the EM vs mesh points . . . . .   | 37   |
| 6. CPU time required to run the IM vs mesh points . . . . .   | 38   |

List of Tables

| Table   | Page |
|---|------|
| I. Comparison of EIGRF Output<br>for First Eigenvalue . . . . .                 | 21   |
| II. Temperature History for $T_q$ . . . . .                                     | 23   |
| III. Time Comparison of<br>Methods in Symmetry . . . . .                        | 29   |
| IV. Time Comparison of<br>Methods in Full Grid . . . . .                        | 30   |
| V. Accuracy Comparison of<br>Methods in Full Grid . . . . .                     | 31   |
| VI. Time Comparison in 3-D<br>in Full Grid . . . . .                            | 36   |
| VII. Data Used in the<br>Determination of $K_1$ . . . . .                       | 39   |
| VIII. Data Used in the<br>Determination of $K_2$ . . . . .                      | 41   |
| IX. Comparison of<br>Accuracy for $N=7$ Cube . . . . .                          | 42   |
| X. Comparison of<br>Accuracy for $5 \times 4 \times 3$ Parallelepiped . . . . . | 43   |

## List of Symbols

### Roman Symbols

|            |   |
|------------|---|
| a          | Length of x side of parallelepiped                                      |
| $a_i$      | The i th multiplying coefficient  |
| b          | Length of y side of parallelepiped                                      |
| B          | Notation for back   |
| c          | Length of z side of parallelepiped                                      |
| $C_v$      | Specific heat   |
| $E_j$      | Overall RMS at time j   |
| F          | Notation for front  |
| h          | Uniform grid size   |
| k          | Thermal conductivity  |
| L          | Length of square slab   |
| n          | Total number of internal mesh points                                    |
| N          | Notation for north  |
| NI         | Internal mesh points (Eq. 40)   |
| P          | Notation for node point   |
| r          | Ratio of time interval<br>to space interval ( $\alpha \Delta t / h^2$ ) |
| $r^*$      | Convient parameter ( $\alpha t / h^2$ )                                 |
| S          | Notation for south  |
| t          | Time  |
| $\bar{t}$  | Dimensionless time ( $t / t^*$ )  |
| $t^*$      | Reference time  |
| T          | Temperature   |
| $u_{ij}^*$ | Exact solution value at point (i,j)                                     |

|          |                                      |
|----------|--------------------------------------|
| $u_{ij}$ | Method solution value at point (i,j) |
| $v^j$    | The j th eigenvector                 |
| W        | Notation for west                    |

Greek Symbols

|             |   |
|-------------|---|
| $\alpha$    | Thermal diffusivity   |
| $\beta_n$   | The y separation constant   |
| $\gamma_0$  | The z separation constant   |
| $\rho$      | Material density  |
| $\phi$      | Transient component of T  |
| $\lambda_i$ | The i th eigenvalue   |
| $\lambda_n$ | The x separation constant   |
| $\Theta$    | Dimensionless temperature $([T_{\alpha,\gamma,z,t}] - T_0) / (T_i - T_0)$ |

Abstract

The eigenvalue method, as presented by T. M. Shih and J. T. Skladany, is evaluated to determine the advantages and disadvantages of the method as compared to fully explicit, fully implicit, and Crank-Nicolson methods. Time comparisons and accuracy comparisons are made in an effort to rank the eigenvalue method in relation to the comparison schemes.

Shih and Skladany's original results are verified by duplicating their efforts with the method. The eigenvalue method is used to solve the parabolic heat equation in multidimensions with transient temperatures. Extensions into three dimensions are made to determine the method's feasibility in handling large geometry problems requiring great numbers of internal mesh points.

The eigenvalue method proves to be slightly better in accuracy than the comparison routines because of an exact treatment, as opposed to a numerical approximation, of the time derivative in the heat equation. It is an unconditionally stable method. It has the potential of being a very powerful routine in solving long transient type problems. The method is not well suited to finely meshed grid arrays or large regions because of the time and memory requirements necessary for calculating large sets of eigenvalues and eigenvectors.

# EVALUATION OF THE EIGENVALUE METHOD IN THE SOLUTION OF TRANSIENT HEAT CONDUCTION PROBLEMS

## I. INTRODUCTION

Heat flow in solid bodies and across their boundaries is a common engineering problem. The solution of these problems can be found by direct calculation, experiment, or by use of numerical techniques. Calculation is the most preferred method, but analytical solutions may not exist in many problems. Experimentation can be expensive and is often inconvenient, if not impossible. Numerical analysis becomes a very good alternative to the problem solver.

At first glance, numerical analysis of heat flow seems "too complicated" and "can't be trusted, as it is only an approximation anyway" (9:viii). This is not a fair opinion, as there are many useful, simple, and very accurate techniques available. Among these are the fully explicit (7:223-251), fully implicit (1:49-53), Crank-Nicolson (8:50-67), and, a fairly new technique in heat transfer, the eigenvalue method (19:409). These do not comprise a complete list of techniques that are available, but, rather, represent broad classes of approaches in discretized solutions.

The eigenvalue method, "which has been used by researchers in structure mechanics, but is relatively unpopular in the heat transfer community" (19:409), can,

indeed, be applied to heat transfer. It is especially well suited to heat transfer problems with long transient times until equilibrium is reached. It differs from the other techniques in that eigenvalues and eigenvectors must be calculated. Once this hurdle, albeit a substantial one, is cleared, the eigenvalue method uses algebraic type equations to arrive at its solutions, thus avoiding time consuming array solving or iterations to converge to correct answers.

#### PROBLEM

A recent (1983) article published by Shih and Skladany (19:409-422) investigated the eigenvalue method to solve transient heat flow problems. The problem investigated in this study is independent verification of their results. Additionally, extensions of the eigenvalue method will be made into three dimensions and the method will be used in an independent problem. Advantages and disadvantages of the method will be identified. The method will be compared to the fully implicit, fully explicit, and Crank-Nicolson methods and, finally, an overall rating will be given in relation to these methods.

#### SCOPE

This study will use cartesian coordinates in two and three dimensions. In addition to the comparison routines, each method's solution will be compared to the exact analytical solution using a least squares analysis. The

eigenvalue method will be extended to three dimensions and compared to a three dimensions version of the fully implicit method.

Cylindrical or spherical coordinates will not be incorporated in the study. It is not the purpose of this study to perform a detailed analysis of the comparison routines. All problems investigated will consider only Dirichlet boundary conditions to insure uniform comparisons. Problems involving heat sources within the solid will not be examined.

#### GENERAL APPROACH

The first objective will be to explain, in detail, what the eigenvalue method (EM) is and why it works. Next, the exact analytical solution to the two dimensional problem presented in the Shih and Skladany article will be verified (19:410).

Armed with this information, the results of the article will be recomputed based upon the sample problem they prepared. The same problem will be solved using the three comparison techniques: fully explicit (EX), fully implicit (IM), and Crank-Nicolson (CN).

Each run will be timed for wall clock times, user time (time actually obeying commands), and central processor unit time (CPU time) (16:473-474). Each method's output will be compared against the exact solution and a least squares analysis will be performed to produce an overall accuracy

for each run (23:1021-1035). All runs will be made using the AFIT VAX 11/780 computer system.

The sample problem will be extended into three dimensions to investigate ease-of-use of the EM in multidimensions and perform an problem independent from that of the article. An additional three dimensional problem will be performed to deliberately remove any symmetry for the purpose of evalutating the EM in these type cases.

Finally, the EM will be closely examined for enhancements, strengths, weaknesses, etc. An overall performance rating, in relation to the comparison routines, will be given.

## II. EXISTING THEORY

Heat conduction in solids has been the object of investigation by engineers for quite some time. The heat diffusion equation is given (21:9-54) as

$$\rho c_v \partial T(x,y,z,t) = k \partial^2 T(x,y,z,t) / \partial x^2 + k \partial^2 T(x,y,z,t) / \partial y^2 + k \partial^2 T(x,y,z,t) / \partial z^2 \quad (1)$$

in three dimensions where

- $\rho$  = material density
- $c_v$  = specific heat
- $k$  = thermal conductivity

All problems examined in this study will be based upon this equation. Shih and Skladany used a two dimensional form of Eq. (1) in their investigations (19:410-413). For simplicity, the following three comparison routines will be examined in just the two dimensions.

### EXPLICIT

One of the simpler numerical techniques is the explicit formulation. The relative ease of setting the explicit method up and the absence of large simultaneous equations are clear advantages of this method (5:91). Its major disadvantage lies in the commonly know problem of stability (19:409; 7:377). This problem links the time step of the

method to the spatial step. This can prove to be prohibitive in large transient type problems.

Use of standard finite differencing techniques will be used in all formulations in this study (7:376-378; 5:59-63). Assuming a forward difference approximation for the time derivative and a central difference approximation for the spatial derivatives, the explicit method can be written as

$$\partial T(x,y,z,t) / \partial t = (T_{i,j}^{t+\Delta t} - T_{i,j}^t) / \Delta t + O(h_t) \quad (2)$$

$$\partial^2 T(x,y,z,t) / \partial x^2 = (T_{i+1,j}^t - 2T_{i,j}^t + T_{i-1,j}^t) / \Delta x^2 + O(h_x^2) \quad (3)$$

$$\partial^2 T(x,y,z,t) / \partial y^2 = (T_{i,j+1}^t - 2T_{i,j}^t + T_{i,j-1}^t) / \Delta y^2 + O(h_y^2) \quad (4)$$

and plugging into the heat equation (1)

$$\partial T / \partial t = \alpha \partial^2 T / \partial x^2 + \alpha \partial^2 T / \partial y^2 \quad (5)$$

where

$$\alpha = k / \rho c_v$$

yields the explicit method algorithm. Placing all the knowns on the right side, the remaining unknown is the temperature at each node for the new time step.

Let  $\Delta x = \Delta y = h$ , then

$$T_{i,j}^{t+\Delta t} = (\alpha \Delta t / h^2) [T_{i+1,j}^t + T_{i-1,j}^t + T_{i,j+1}^t + T_{i,j-1}^t - 4T_{i,j}^t] + T_{i,j}^t \quad (6)$$

This algorithm can be used to calculate temperatures at any nodal point  $(i,j)$  at any time  $t$ , provided the  $\Delta t$  is sufficiently small to meet the stability requirement.

To insure a bounded answer, but not necessarily accuracy (15), the condition

$$\alpha \Delta t / h^2 \leq 1/4 \quad (7)$$

must be met (7:379-380). This method has truncation errors with order  $O(h\Delta t) + O(h^2/h^2)$  (5:59-63).

### IMPLICIT

To avoid the serious stability criteria of the explicit method, the engineer can turn to an implicit formulation. The main principle of this scheme is that the temperature at each point is found at all nodes simultaneously using the temperature values from the previous time step (17:58). This also gives rise to its disadvantage, in that large systems of simultaneous equations must be solved to arrive at a solution. For finely meshed problems with long transient times this can become a significant problem.

Formulation of the implicit scheme is similar to that of the explicit scheme except the spatial derivatives are evaluated at the new time step (evaluated at the current time step in explicit).

The time derivative will remain unchanged and the spatial derivatives will become (5:104; 1:49-53)

$$\partial^2 T / \partial x^2 = (T_{i+1,j}^{t+1} - 2T_{i,j}^{t+1} + T_{i-1,j}^{t+1}) / \Delta x^2 + O(h_x^2) \quad (8)$$

$$\partial^2 T / \partial y^2 = (T_{i,j+1}^{t+1} - 2T_{i,j}^{t+1} + T_{i,j-1}^{t+1}) / \Delta y^2 + O(h_y^2) \quad (9)$$

Plugging Eqs. (2), (8), and (9) into Eq. (5) and then rearranging all the unknowns on one side yields

$$T_{i,j}^t = - (\alpha \Delta t / h_x^2 h_y^2) [ T_{i+1,j}^{t+1} + T_{i-1,j}^{t+1} + T_{i,j+1}^{t+1} + T_{i,j-1}^{t+1} - 4T_{i,j}^{t+1} ] + T_{i,j}^{t+1} \quad (10)$$

This method is well adapted to the computer because it is unconditionally stable, but it does generate the large sets of simultaneous equations (11:455). This method has truncation error on order of  $O(h_x) + O(h_{xy}^2)$  (5:59-63).

### CRANK-NICOLSON

An attempt to avoid numerical instability led to the implicit method. The method is unconditionally stable, but the method still had truncation errors of  $O(h_t) + O(h_{xy}^2)$  (5:59-63). This can be improved to  $O(h_t^2) + O(h_{xy}^2)$  by approximating the time derivative with a central difference (11:455). This method, it turns out, has problems with stability (11:455). Crank and Nicolson suggested to overcome this by representing the derivative with an average of a forward difference and a backward difference, each weighted by 1/2 (8:50; 11:455-457). If

$$\alpha \Delta t / h_{xy}^2 = r \quad (11)$$

then the Crank-Nicolson scheme is

$$T_{ij}^{t+1} = (1-2r)/(1+2r) T_{ij}^{t+1} + \left\{ r / [2(1+2r)] \right. \\ \left. [ T_{i+1,j}^{t+1} + T_{i-1,j}^{t+1} + T_{i,j+1}^{t+1} + T_{i,j-1}^{t+1} \right. \\ \left. + T_{i+1,j}^t + T_{i-1,j}^t + T_{i,j+1}^t + T_{i,j-1}^t ] \right\} \quad (12)$$

This method also generates large systems of simultaneous equations. The advantage comes from the increase in the accuracy gained by the  $O(h_t^2)$  term.

### III. NEW THEORY

#### THE EIGENVALUE METHOD

The eigenvalue method (EM) is not really a "new" theory or discovery. Shih and Skladany suggest in their article that this method has been used extensively in structure mechanics (19:405). The use of eigenvalues to solve systems of equations has been studied in math text books for quite some time. It is the application of the EM to solutions of the heat equation that is fairly new.

#### WHAT IS THE EM

Solving problems numerically requires the division of the region in question into small grids. The intersection of the grid lines (called nodal points), are, typically, the points of interest in the study. If these grids are sufficiently small, the exact behavior under study can be quite accurately approximated.

Each node point is an unknown value and is generally represented by a governing equation. A system of equations such as these can be solved simultaneously to find these unknowns. It takes  $n$  equations to find  $n$  unknowns. Provided the equations can be found, all that remains is the selection of a technique to solve them. Several techniques exist to do just that; among them is the EM.

### WHY DOES THE EM WORK?

For clarity, the EM is applied to the transient heat equation. This application demonstrates the method and why it is a valid numerical technique.

The transient parabolic heat equation in three dimensions appears as

$$\rho c v \partial T / \partial t = k \left[ \partial^2 T / \partial x^2 + \partial^2 T / \partial y^2 + \partial^2 T / \partial z^2 \right] \quad (13)$$

Using standard central differences for the spatial derivatives (7:376-378)

$$\begin{aligned} \rho c v \partial T / \partial t = k & \left[ (T_{i+1,j,k} - 2T_{i,j,k} + T_{i-1,j,k} / \Delta x^2) \right. \\ & + (T_{i,j+1,k} - 2T_{i,j,k} + T_{i,j-1,k} / \Delta y^2) \\ & \left. + (T_{i,j,k+1} - 2T_{i,j,k} + T_{i,j,k-1} / \Delta z^2) \right] \quad (14) \end{aligned}$$

and, again, let  $\Delta x = \Delta y = \Delta z = h$ .

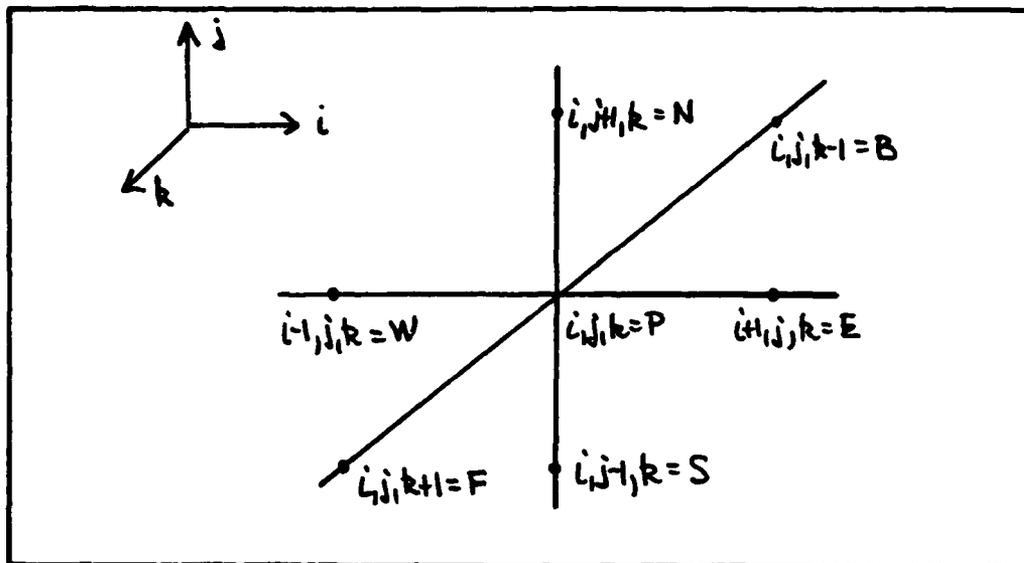


Figure 1. Notation Convention for Discretizations.

Figure 1 depicts the notation convention used in this report for all discretization schemes. Using this notation, Eq. (14) can be written as

$$\rho c v \Delta T_p / \Delta t = (k/h^2) [T_E + T_W + T_N + T_F + T_S + T_B - 6T_P] \quad (15)$$

where

$$p = 1, 2, 3, \dots, n.$$

#### DERIVATION OF THE METHOD

Borrowing mainly from the Shih and Skladany report (19:409-422), the following is the derivation of the eigenvalue method. Equation (15) can be expressed in matrix

form as

$$[C]\{\dot{T}\} = [K]\{T\} + \{F\} \quad (16)$$

where "[C] is an nxn identity matrix multiplied by  $\rho h C_V$ , [K] is an nxn sparse conductance matrix whose diagonal elements are  $-4k/h^2$  ( $-6k/h$  in three dimensions) (19:411). The vector  $\{F\}$  is of length n and incorporates the boundary conditions.

In general, all matrices have n characteristic eigenvalues and associated eigenvectors (14:345). Most discretization schemes do not make use of these. Along with the appropriate multiplying constants determined from boundary conditions, these eigenvalues and eigenvectors can be used to solve the system of simultaneous equations from which they came (14:824). This is the basis of the EM and why it works.

Use of some simple matrix algebra reveals the core of the method. If Eq. (16) is pre-multiplied by  $[C]^{-1}$  to yield (again, following Shih and Skladany) (19:411)

$$[C]^{-1}[C]\{\dot{T}\} = [C]^{-1}[K]\{T\} + [C]^{-1}\{F\} \quad (17)$$

and define

$$[C]^{-1}[K] = [D] \quad (18)$$

$$[C]^{-1}\{F\} = \{b\} \quad (19)$$

to get

$$\{\dot{T}\} = [D]\{T\} + \{G\} \quad (20)$$

Now, let

$$\{T\} = \{\phi(t)\} + \{T_F\} \quad (21)$$

where  $\{\phi(t)\}$  is some arbitrary function of time and  $\{T_F\}$  is independent of time and represents the steady-state solution. The derivative of Eq. (21) with respect to time gives

$$\{\dot{T}\} = \{\dot{\phi}(t)\} \quad (22)$$

and plugging Eqs. (21) and (22) into Eq. (20)

$$\{\dot{\phi}(t)\} = [D]\{\phi(t)\} + [D]\{T_F\} + \{G\} \quad (23)$$

and if

$$\{T_F\} = -[D]^{-1}\{G\} \quad (24)$$

then Eq. (23) becomes just

$$\{\dot{\phi}(t)\} = [D]\{\phi(t)\} \quad (25)$$

which is a set of "first-order, linear, ordinary differential equations" (19:411). Appendix B details the solution of Eq. (25) as applied to a system of two equations (without loss of generality). Based upon this and assuming the solution of Eq. (25) to be

$$\{\phi_i(t)\} = \sum_i \{c_i\} \exp(\{\lambda_i\}t) \quad (26)$$

where  $\{c_i\}$  is the set of multiplying constants yet to be determined and  $\{\lambda_i\}$  are the eigenvalues of the system. It turns out the multiplying constants are actually a product of the eigenvectors and another constant determined by the boundary conditions of the particular problem. Once the eigenvalues, eigenvectors, and the multiplying constants are determined, the Eq. (25) can be solved. These results are used in Eq. (21) to find the temperature at each nodal point. In matrix form the EM can be written

$$\{T_i\} = \{T_F\} + \sum_{i,j} \{a_i\} \{v_i^j\} \exp(\{\lambda_i\}t) \quad (27)$$

where

$$i = j = 1, 2, 3, \dots, n.$$

and  $a_i$  is the  $i$ -th multiplying coefficient determined by boundary conditions and  $v_i^j$  is the  $j$ -th component of the  $i$ -th eigenvector (see List of Symbols).

## ADVANTAGES AND DISADVANTAGES

From the previous derivation and the Shih and Skladany article several advantages and disadvantages seem possible.

Among the advantages are

1. Better accuracy; no time truncation error.
2. Efficient use with long transients.
3. Unconditionally stable method.
4. Rapid solution; few eigenvectors needed.

Two possible disadvantages could be

1. Difficulty in finding and using good eigenvalue/vector routines.
2. Time consuming in fine mesh space problems.

The results of this study will verify or contradict these and bring to light any others not mentioned.

#### IV. APPLICATION TO SPECIFIC PROBLEMS

One of the main purposes of this study is to verify results achieved by Shih and Skladany in their article. They presented a two dimensional sample problem (19:412). For the purpose of brevity, they only examined this problem by taking full advantage of the symmetry involved. This study does the same and will also examine the problem with the assumption that not all problems will be symmetric.

##### THE TEST PROBLEM

Consider a square slab of length,  $L$ , on a side with constant temperature,  $T_0$ , on each of the sides. Further, let the region be governed by

$$\rho c v \partial T / \partial t = k \partial^2 T / \partial x^2 + k \partial^2 T / \partial y^2 \quad (28)$$

subject to the boundary conditions

$$T(0, y, t) = T_0$$

$$T(L, y, t) = T_0$$

$$T(x, 0, t) = T_0$$

$$T(x, L, t) = T_0$$

and the initial condition

$$T(x, y, 0) = T_i$$

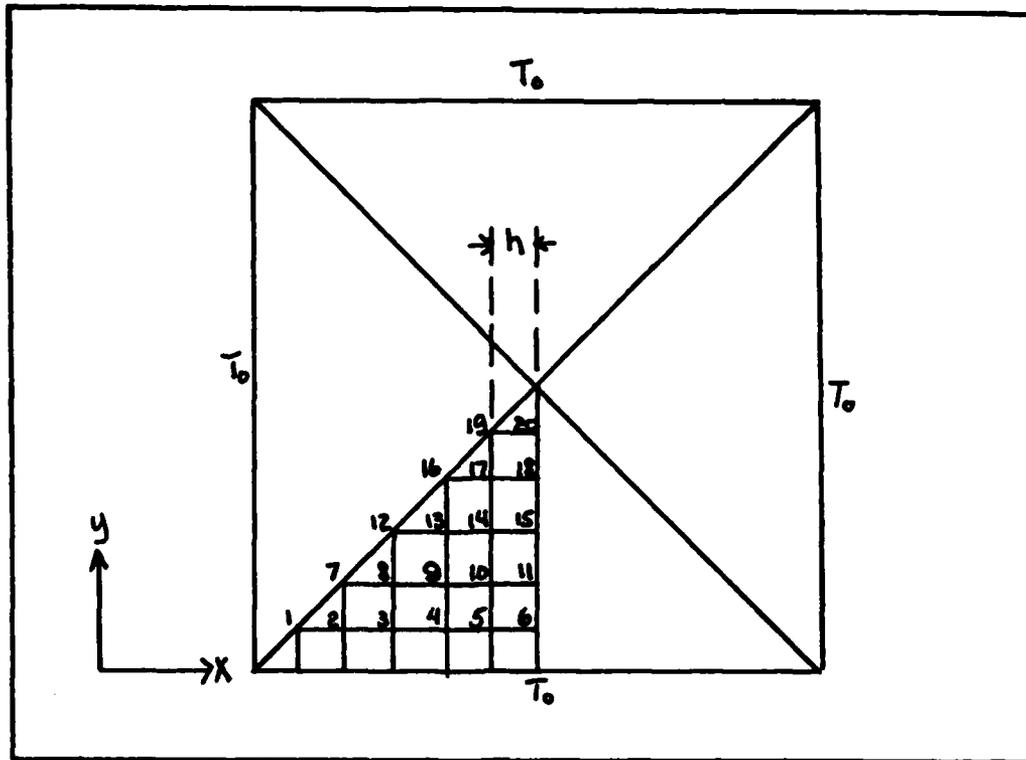


Fig. 2. Domain for 2-D transient heat conduction problem.

Figure 2 shows graphically the sample problem. Shih and Skladany maintain that the temperature at the node point in the center is equal to the temperature at  $T_{20}$  (19:412). It is not necessary to make this approximation to assure  $h^2$  accuracy. A more accurate formulation is to make the temperature at the center separate and add an additional equation to the resulting system.

The analytical solution to Eq. (28) can be found in Appendix A by removing the  $z$  dependence. The normalized solution will then be

$$\Theta(x, y, t) = \Theta_1(x, t) \Theta_2(y, t) \quad (29)$$

where

$$\Theta_1(x,t) = \sum_{n=1}^{\infty} (4/n\pi) \exp(-n^2\pi^2 kt/L^2) \sin(n\pi x/L) \quad (29a)$$

and

$$\Theta_2(y,t) = \sum_{n=1}^{\infty} (4/n\pi) \exp(-n^2\pi^2 kt/L^2) \sin(n\pi y/L) \quad (29b)$$

with

$$n = 1, 3, 5, \dots$$

for both  $\Theta_1$  and  $\Theta_2$  .

For simplicity let (19:412)

$$\alpha t^* / h^2 = 1 \quad (30)$$

The reader is advised to refer to the list of symbols found at the beginning of this report in order to keep track of the normalized and real values. For generality, this problem is done with normalized values.

In the interest of brevity, only a few of the 20 differential equations generated by Eq (15) (2-D version) are listed

$$\partial T_1 / \partial \bar{E} = 2T_0 - 4T_1 + 2T_2 \quad (31)$$

.....

$$\partial T_{17} / \partial \bar{E} = T_{14} + T_{16} + T_{18} + T_{19} - 4T_{17} \quad (32)$$

$$\partial T_{18} / \partial \bar{E} = T_{15} + 2T_{17} + T_{20} - 4T_{18} \quad (33)$$

$$\partial T_{19} / \partial \bar{E} = 2T_{17} + 2T_{20} - 4T_{19} \quad (34)$$

$$\partial T_{20} / \partial \bar{E} = T_{18} + 2T_{19} - 3T_{20} \quad (35)$$

These 20 equations represent a matrix whose coefficients are used to find the 20 eigenvalues and eigenvectors for use in the EM.

This brings up a major criticism of this method. The matrix formed by the above is not symmetric, banded, and is not positive definite. These are not desirable characteristics of a general matrix. This matrix required the use of the IMSL routine EIGRF (10:EIGRF-1-EIGRF-2) to find the eigenvalues and eigenvectors. This routine is particularly difficult to use and did not interface well with the VAX 11/780 computer.

TABLE I

Comparison of EIGRF Output for First Eigenvalue.

|             | STUDY   | ARTICLE | RATIO  |
|-------------|---------|---------|--------|
| EIGENVALUE  | -0.1399 | -0.1400 | 0.9999 |
| COEFFICIENT | -821.28 | -832.04 | 0.8812 |
| v(1,1)      | 0.0269  | 0.0267  | 1.0075 |
| v(1,2)      | 0.0520  | 0.1516  | 1.0078 |
| v(1,3)      | 0.0734  | 0.0729  | 1.0069 |
| v(1,4)      | 0.0898  | 0.0891  | 1.0090 |
| v(1,5)      | 0.1000  | 0.0993  | 1.0070 |
| v(1,6)      | 0.1034  | 0.1027  | 1.0068 |
| v(1,7)      | 0.1003  | 0.0996  | 1.0070 |
| v(1,8)      | 0.1416  | 0.1406  | 1.0071 |
| v(1,9)      | 0.1731  | 0.1719  | 1.0070 |
| v(1,10)     | 0.1927  | 0.1913  | 1.0073 |
| v(1,11)     | 0.1994  | 0.1979  | 1.0076 |
| v(1,12)     | 0.1998  | 0.1984  | 1.0075 |
| v(1,13)     | 0.2440  | 0.2423  | 1.0070 |
| v(1,14)     | 0.2714  | 0.2695  | 1.0071 |
| v(1,15)     | 0.2806  | 0.2786  | 1.0072 |
| v(1,16)     | 0.2976  | 0.2955  | 1.0071 |
| v(1,17)     | 0.3304  | 0.3281  | 1.0070 |
| v(1,18)     | 0.3410  | 0.3386  | 1.0071 |
| v(1,19)     | 0.3654  | 0.3628  | 1.0073 |
| v(1,20)     | 0.3747  | 0.3720  | 1.0073 |

Table I lists a partial comparison of the output of the IMSL routine EIGRF. This is only a sample of the complete output and clearly shows the results (19:414) Shih and Skladany achieved are correct. The eigenvectors have an average error  $\pm 0.728\%$ . The eigenvalues were the same as the article (within 4 decimal places). The coefficient is a little different from the Shih and Skladany coefficient because of the slight differences in the system of eigenvectors. Despite the difference, the errors are compensating and account for a negligible difference in the temperature histories. The coefficients were calculated using the initial condition and solving the resulting matrix by the IMSL routine LEQT2F (10:LEQT2F-1-LEQT2F-3; 24). All IMSL routines used in this study can be found in Appendix C.

#### RECURSION FORMULAE

With the eigenvalues, eigenvectors, and multiplying coefficients known, the EM can be used to solve the problem (19:413). The nodal point temperature histories can be found by

$$T_i(\bar{t}) = T_0 + \sum_{i=1}^n a_i v_i \exp(\lambda_i t) \quad (36)$$

Table II presents a partial comparison of the results of Eq. (36). Temperatures at point T<sub>9</sub> are used as a sample of the entire grid.

TABLE II

Temperature History for  $T_9$  .

| TBAR | STUDY   | EXACT   | RATIO |
|------|---------|---------|-------|
| 1.0  | 35.211  | 32.248  | 1.092 |
| 2.0  | 70.714  | 69.683  | 1.015 |
| 3.0  | 95.422  | 94.984  | 1.005 |
| 4.0  | 113.134 | 112.830 | 1.003 |
| 5.0  | 126.542 | 126.216 | 1.003 |
| 6.0  | 137.181 | 136.786 | 1.003 |
| 7.0  | 145.913 | 145.447 | 1.003 |
| 8.0  | 153.244 | 152.720 | 1.003 |
| 9.0  | 159.485 | 158.919 | 1.003 |
| 10.0 | 164.845 | 164.254 | 1.004 |

The above table shows the comparison at point  $T_9$  with  $r^* = 1$  as compared to the exact solution at the same node point. When plotted as a graph, these values appear in Figure 3 on the following page. From Table II and Fig. 3, it is seen that the eigenvalue method compares very well with the exact solution after  $\bar{t}$  of 2 or more. After this time the eigenvalue method was within  $\pm 0.3\%$  of the exact answer. Before  $\bar{t} = 2$  the method differed considerably. Shih and Skladany offer an explanation for this (19:417).

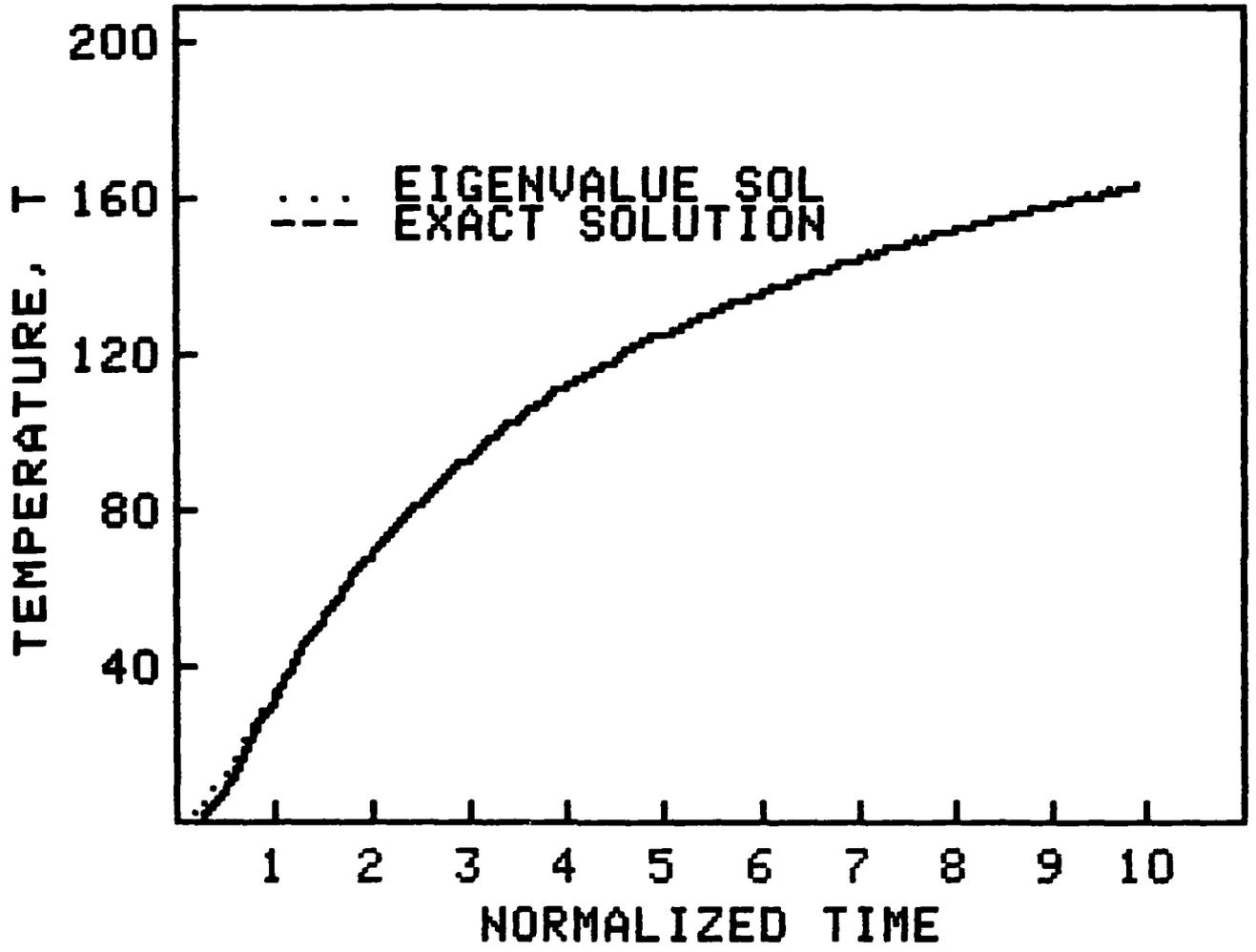


Figure 3. History of temperature at nodal point  $T_9$ .

"Since initially the heating rate  $\partial T/\partial t$  and the change of the heating rate  $\partial^2 T/\partial t^2$  are large, and since  $\partial T/\partial t$  is related to the space derivatives by

$$\partial T/\partial t = \alpha \nabla^2 T$$

it follows that initially the value of  $\nabla^4 T$  must be large because

$$\partial^2 T/\partial t^2 = \alpha \nabla^2 (\partial T/\partial t) = \alpha^2 \nabla^4 T$$

Now we already know that the fourth derivative is embedded in the leading truncated term in the second order accurate finite difference approximation."

This explanation also implies the truncation error will be large in the initial heating in all three schemes.

Equation (36) was used as the recursion formula for the EM. Using the notation of Figure 1, the following recursion formulae were used in the study as the comparison routines.

The explicit method used

$$T_p^{t+1} = (\alpha \Delta t / h^2) [T_S^t + T_W^t + T_E^t + T_N^t - 4T_p^t] + T_p^t \quad (37)$$

The implicit method used

$$T_p^t = -(\alpha \Delta t / h^2) [T_E^{t+1} + T_W^{t+1} + T_N^{t+1} + T_S^{t+1}] + [1 + 4(\alpha \Delta t / h^2)] T_p^{t+1} \quad (38)$$

and the Crank-Nicolson used

$$T_p^{th} = \left[ \frac{(1-2\tau)}{(1+2\tau)} \right] T_p^t + \left\{ \left[ \frac{\tau}{2(1+2\tau)} \right] \right. \\ \left. \left[ T_E^{th} + T_W^{th} + T_N^{th} + T_S^{th} + T_E^t + T_W^t + T_N^t + T_S^t \right] \right\} \quad (39)$$

where

$$\tau = \alpha \Delta t / h^2$$

### ERROR EVALUATIONS

An article written by Towler and Yang was the basis for selection of an error routine (23:1021-1024). The desire for some overall way of determining a method's accuracy was used as a criteria in the selection of the Root Mean Squared (RMS) method. Stated simply, this finds the difference between the temperature at each node and the exact temperature, squares each point's differences, adds up all of these, divides by the total number of inner node points, and, finally, takes the square root of the answer. Written mathematically as (23:1023)

$$E_j = \sqrt{\frac{\left[ \sum_{i=1}^N (u_{ij} - u_{ij}^*)^2 \right]}{NT}} \quad (40)$$

This provides a benchmark for each routine to base accuracy evaluations upon.

The exact solution found in Appendix A is in the form of a summation of terms involving exponentials, sines, and cosines. When using normalized time ( $t = 1, 2, 3, \dots$ ) the exponential term will drive the magnitude of the individual term's contribution. Thus, when  $n$ ,  $m$ , and  $o$  become large, the term will be small and add little to the summation. When using normalized time it is found that only 15 terms in the sum are required for convergence to  $10^{-10}$  accuracy. This degree of accuracy is seldom necessary in light of the accuracy of such measured things as thermal conductivity, specific heat, etc. To retain a reasonable amount of accuracy of 3 or 4 significant digits, only 11 terms are required in the summation. This translates into significant time savings in the computation of the exact solution for comparison purposes.

#### RESULTS OF 2-D PROBLEM

The sample problem in Shih and Skladany's article was symmetric in the x-y plane (19:413). For this reason, they elected to take full advantage of symmetry and, thus, make the computational domain fairly small (a 20X20 matrix). Full symmetry in everyday engineering problems is idealistic (22:127). Without symmetry, the domain will be analyzed by a full grid with pitch  $h$ . This study ran the test problem both ways. Symmetry was used for the initial verification.

The same problem was then solved using the full grid (same h) without taking advantage of the symmetry. This approach brought out some interesting observations.

Taking advantage of symmetry produces a matrix that is non-symmetric, non-banded, and is non-positive definite. This type matrix required the use of a more time consuming routine (EIGRF). It also required more core memory as the matrix must be stored in its full form. The full grid analysis produces a banded, symmetric, non-positive definite matrix. This saved central processor time (CPU time) and core memory as only the lower co-diagonals needed to be stored to compute the eigenvalues/vectors.

Shih and Skladany did not do a time comparison to determine if the EM approach was faster (and therefore, less expensive) on computers than the other types of solving schemes. This was accomplished in this study using the VAX 11/780 and the UNIX command TIME (16:473-474). This command returned several pieces of information including wall clock time and the CPU time.

Taking advantage of TIME and the RMS error routine, the four methods were run in symmetry and, then, in full grid forms. The results were then compared to the exact solution.

## TIME COMPARISONS

The following tables represent the time comparisons of the specific methods. The CPU times are accurate to within 1/10 second and the wall clock times are accurate to within 1/60 second (16:473-374). The wall clock times may not be consistent because runs made during different times of the day took more or less wall clock time according to the computer's load. The CPU times will be consistent as this is the time the central processor takes to execute the program.

TABLE III

Time Comparison of Methods in Symmetry.

| METHOD     | CPU (SEC) | WALL (HR:MIN:SEC) |
|------------|-----------|-------------------|
| EIGENVALUE | 0.8       | 0:00:42           |
| EXPLICIT   | 0.7       | 0:02:13           |
| IMPLICIT   | 4.2       | 0:16:34           |
| CN         | 6.4       | 0:23:56           |

Here it is easy to see the EM is very comparable to the explicit method and is clearly faster than the last two methods. All programs were run to achieve similar accuracies (three significant digits the same).

TABLE IV

Time Comparison of Methods in Full Grid.

| METHOD     | CPU (SEC) | WALL (HR:MIN:SEC) |
|------------|-----------|-------------------|
| EIGENVALUE | 40.1      | 3:04:19           |
| EXPLICIT   | 15.1      | 1:44:47           |
| IMPLICIT   | 6.1       | 0:41:41           |
| CN         | 26.4      | 1:29:00           |

The value of the use of the full grid approach becomes apparent. When comparing Tables III and IV the CPU times do not seem consistent. The explicit time in Table IV can be explained by realizing the time restriction in Eq. (7) must be satisfied. In this case  $\Delta \bar{t} = 1/1000$  to achieve the necessary accuracy. This requires 1000 incremental calculations to get to  $\bar{t} = 1$  ! The implicit time is less than explicit because it does not require this sort of time step. To achieve the same accuracy the  $\Delta \bar{t} = 1/50$  . The CN scheme takes more time in both tables and is therefore consistent.

The problem of having to find the eigenvalues and eigenvectors starts to become clear here. Finding eigenvalues/vectors in full grid results in nxn matrices being formed. The full grid sample used 121 internal mesh points (11X11 grid, 121X121 matrix!). In full grid mode the method now takes 4 times as long to run. In this case all

of the other methods are preferable to the EM; just considering the time required to execute.

ACCURACY COMPARISONS

The following table represents the accuracy comparison of the specific methods. Only the full grid form of the problem was compared because the errors are symmetric as well as the temperatures. Values given are in °K. All problems were run with the same time step of  $\Delta\bar{t} = 1/50$ .

TABLE V

Accuracy Comparison of Methods in Full Grid.

| METHOD   | TBAR |      |      |      |      |      |      |      |      |      |
|----------|------|------|------|------|------|------|------|------|------|------|
|          | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    | 10   |
| EIGEN    | 2.24 | 1.42 | 1.14 | 0.93 | 0.73 | 0.56 | 0.42 | 0.32 | 0.23 | 0.17 |
| EXPLICIT | 6.86 | 4.34 | 3.49 | 2.84 | 2.23 | 1.71 | 1.30 | 0.98 | 0.72 | 0.52 |
| IMPLICIT | 7.23 | 4.63 | 3.73 | 2.98 | 2.29 | 1.71 | 1.26 | 0.90 | 0.63 | 0.42 |
| CN       | 7.20 | 4.59 | 3.73 | 3.06 | 2.44 | 1.92 | 1.50 | 1.16 | 0.89 | 0.68 |

Figure 4 on the following page is a plot of Table V. This figure dramatically shows the improved accuracy of the eigenvalue method over the more conventional methods.

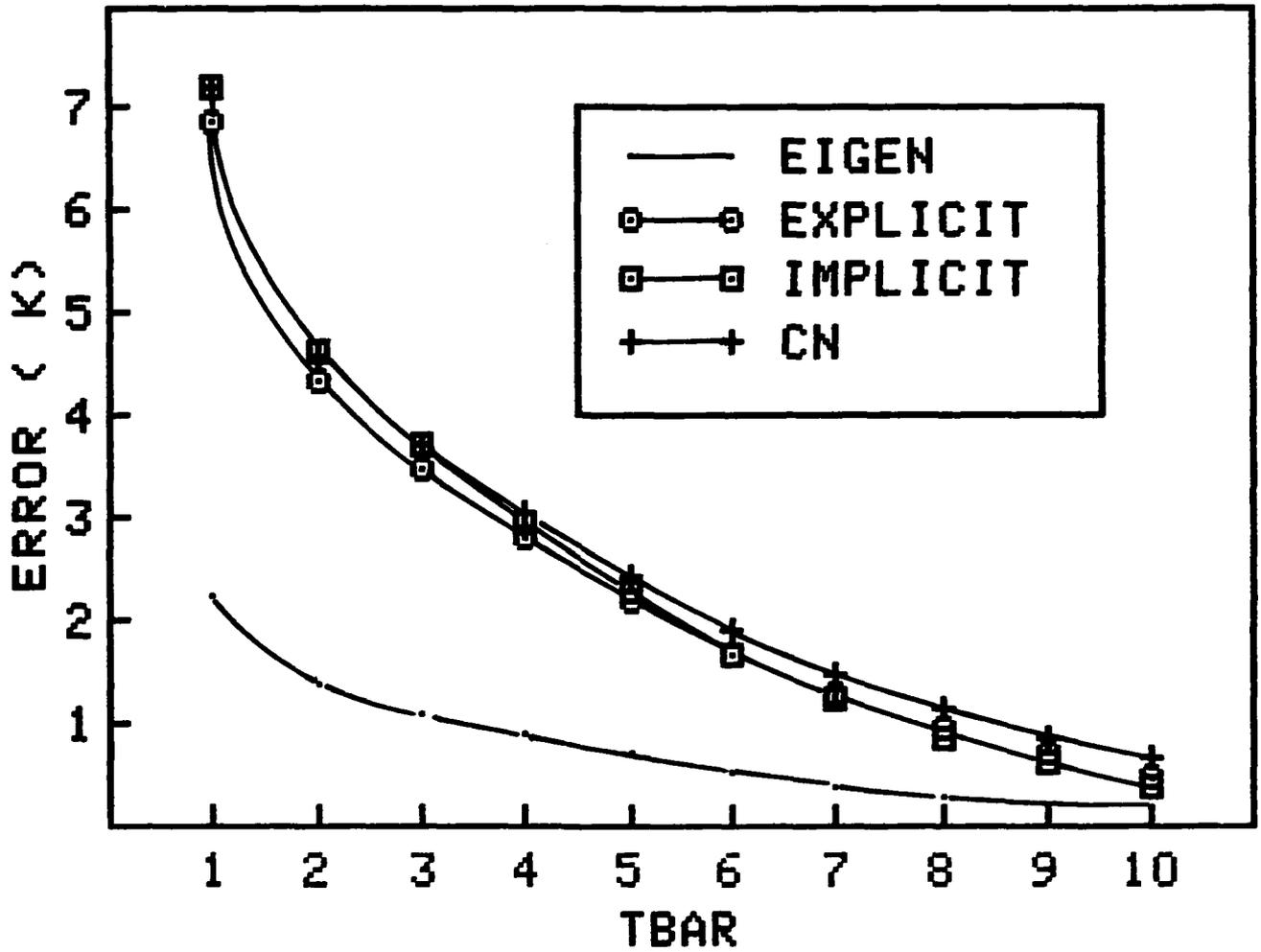


Figure 4. Accuracy comparison of methods in full grid.

### EXTENSION OF SAMPLE PROBLEM INTO 3-D

A literature search of the numerical techniques in solving the heat equation revealed very few sources that go beyond a two dimensional case (17:45-48; 4:142-144; 3:162-171). It is, therefore, instructive to examine a problem in three dimensions.

Several questions arise immediately when considering the three dimensional case.

1. Does an analytical solution exist?
2. What boundary conditions are appropriate?
3. Does the method accuracy change in 3-D?
4. How much more time is required to run?

Answers to these questions are supplied in the following sections.

### THE THREE DIMENSION PROBLEM STATEMENT

The simplest case to study is a cube of dimension  $L$  on a side. A more general problem also worth considering is a parallelepiped (a box) with dimensions as show in Figure 4.

The general case can be easily modified by making the lengths  $a$ ,  $b$ , and  $c$  all equal to  $L$  and the cube case is back. The program written took advantage of this, thus saving work and time.

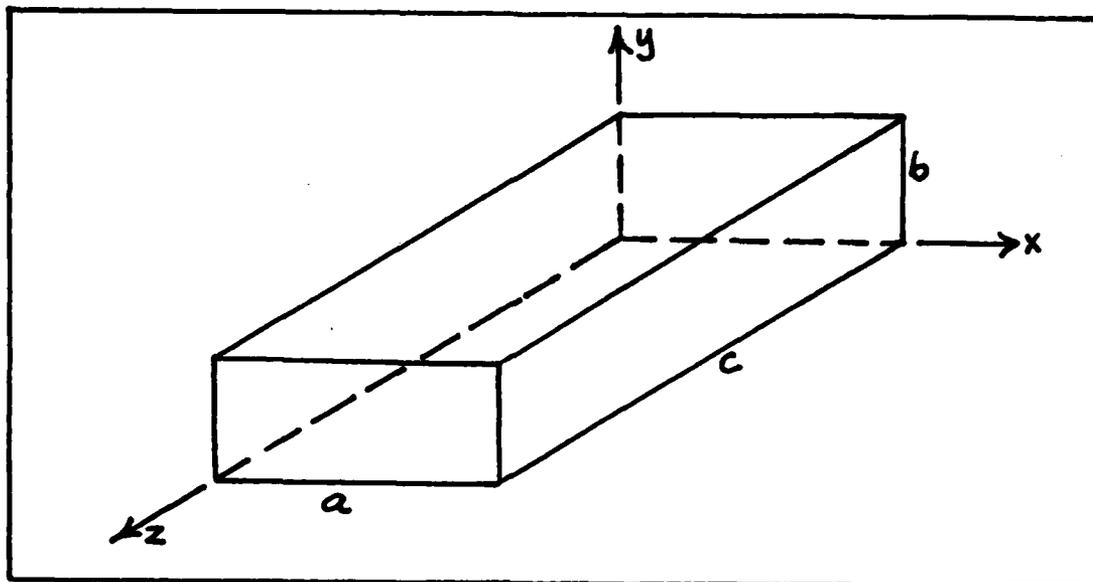


FIGURE 4. Three dimension test problem.

All the surfaces of the box will be at the same  $T_0 = 200$  °K. All the interior points are at 0 °K initially. Therefore, the governing equation, boundary conditions, and initial condition are

$$\partial T / \partial t = k \left[ \partial^2 T / \partial x^2 + \partial^2 T / \partial y^2 + \partial^2 T / \partial z^2 \right] \quad (41)$$

where

$$T(0, y, z, t) = T_0$$

$$T(a, y, z, t) = T_0$$

$$T(x, 0, z, t) = T_0$$

$$T(x, b, z, t) = T_0$$

$$T(x, y, 0, t) = T_0$$

$$T(x, y, c, t) = T_0$$

$$T(x,y,z,0) = T_i$$

Using the same normalizations as the 2-D case and referring to Appendix A, the solution to Eq. (41) is given by Eq. (A-41). Indeed, the boundary conditions are appropriate and there is an analytical solution to compare the methods against.

Time restrictions during the study only allowed for the implicit method to be developed in three dimensions as a comparison routine to the exact and eigenvalue methods. Little is lost here because the 2-D results extend into the 3-D case.

Recalling that the number of internal mesh points determines the n dimension of the eigenvector/value solving routine, a balance must be met. Enough mesh points must be supplied to insure accuracy (not necessarily stability) (15) while at the same time too many mesh points cause the problem to run extremely long. In the eigenvalue case, a cube with 8 divisions (7 internal points) on a side was the maximum number the VAX computer could handle before running out of core memory. The core memory requires  $n^2$  bytes (10) available for the eigenvalue method to use even in full grid form (banded storage mode).

The EM is not an ideal method if computer memory space is at a premium. This is another criticism of this method. Shih and Skladany did not explore past the 20X20 matrix they used and this aspect of the eigenvalue method was not made apparent. This could present a definite problem to the user

if he is trying to solve a heat problem for something large, such as a nuclear reactor core.

### THREE DIMENSIONAL TIME RESULTS

The three dimension time runs were very instructive. The fact that 3-D is used is not important. What the 3-D runs bring out is the use of many node points. It is in this that the true nature of the eigenvalue time requirements become apparent.

Both the EM and IM were run for cubes with  $L = 3, 4,$  and 5. The times for each run are listed below

TABLE VI

Time Comparison in 3-D in Full Grid.

| METHOD | L=3 | L=4 | L=5  |
|--------|-----|-----|------|
| EM     | 3.0 | 4.9 | 40.1 |
| IM     | 1.9 | 3.1 | 6.1  |

The time for the EM at  $L = 5$  seemed extraordinarily large. An additional run verified the first number. A plot of several runs at differing numbers of mesh points should better describe the behavior of the method's time to execute requirements.

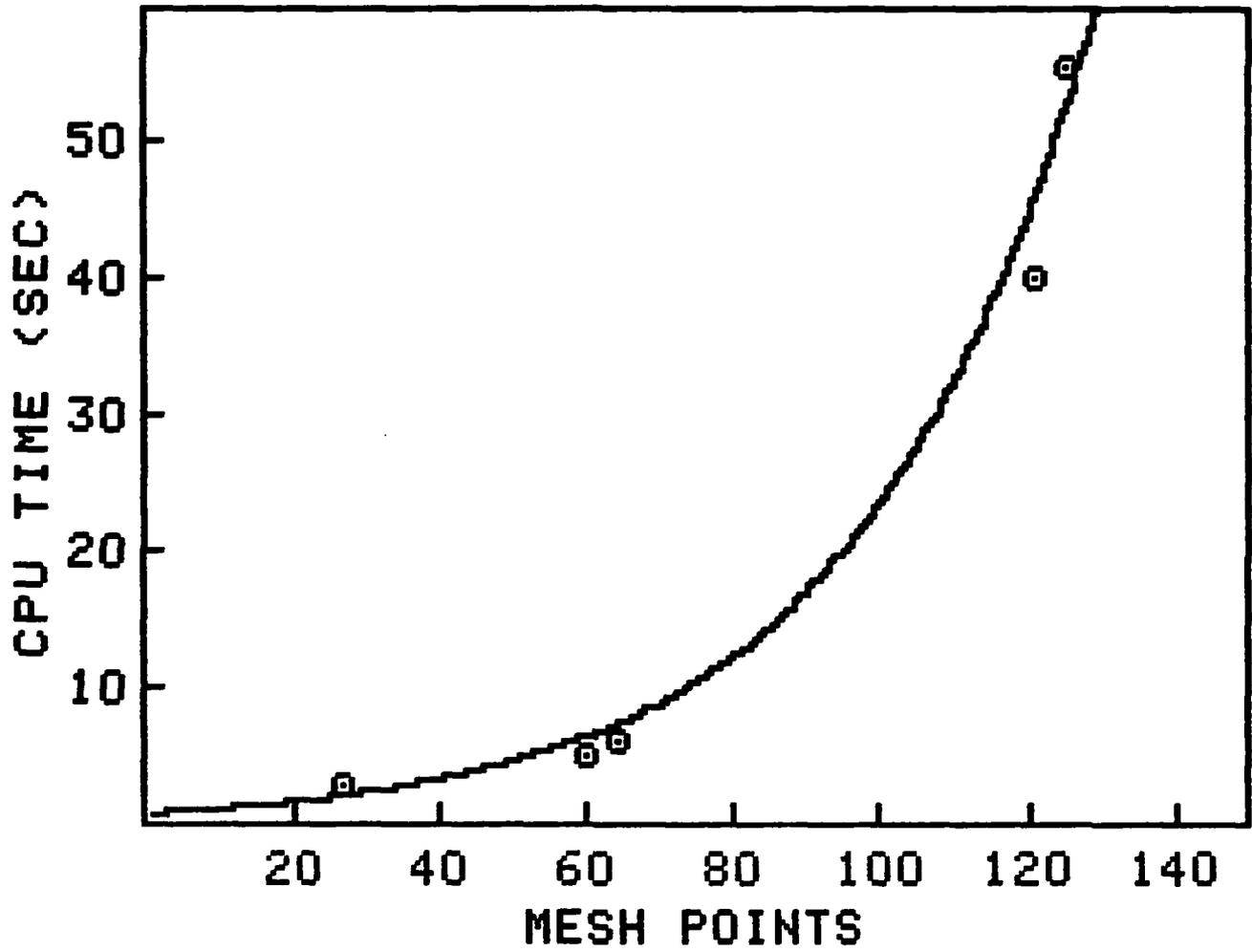


Figure 5. CPU time required to run the EM vs mesh points.

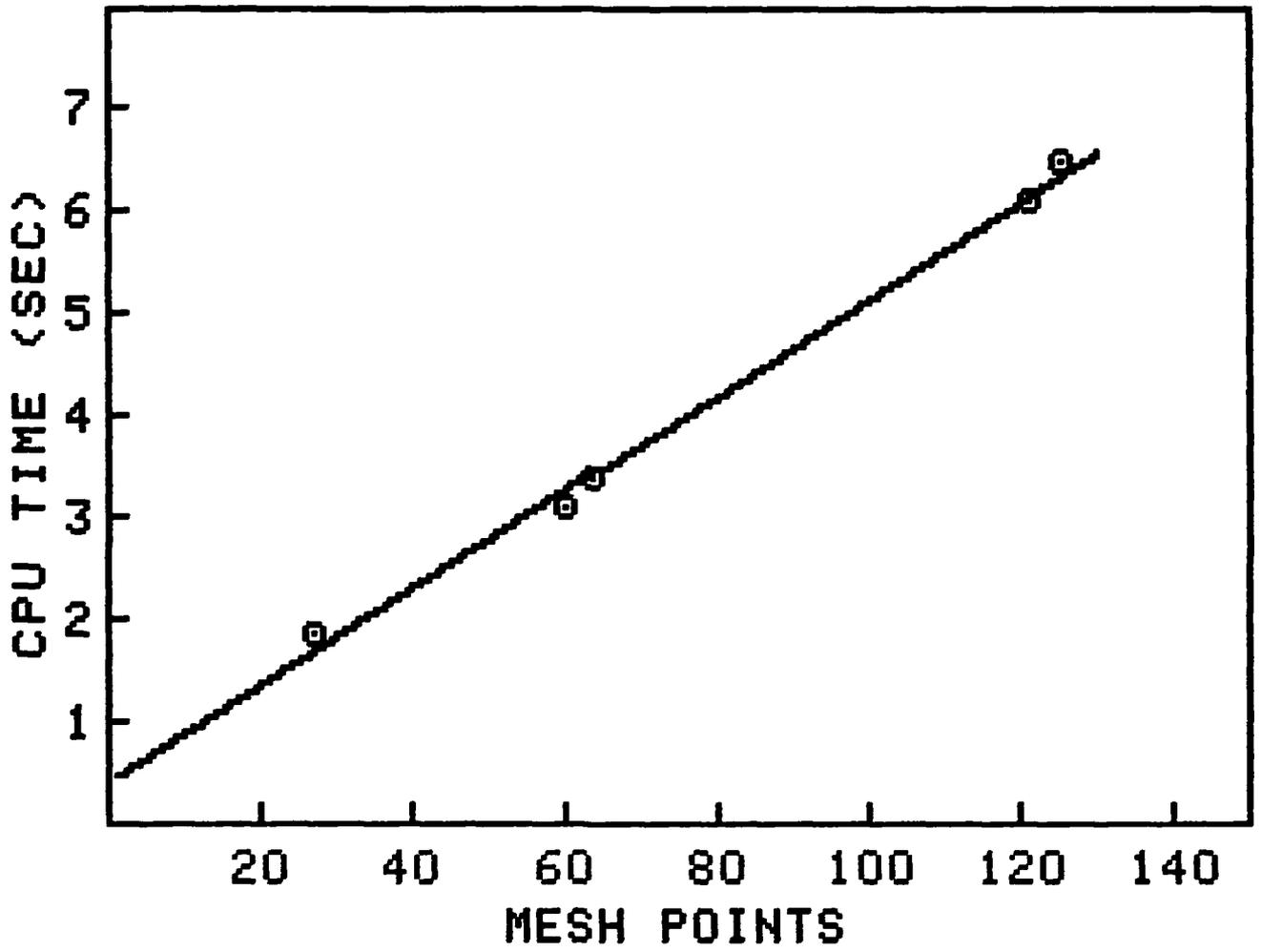


Figure 6. CPU time required to run the IM vs mesh points.

Figures 5 and 6 are the plots of the EM and IM CPU time requirements, respectively. It appears the EM is exponential in nature with respect to the number of mesh points while the IM is simply linear. The curves drawn in the figure represent the mean value of the data points.

The first curve (Fig. 5) was drawn by developing the following data:

TABLE VII

Data Used for the Determination of K1.

| GRID  | POINTS | TIME (SEC) |
|-------|--------|------------|
| 3X3X3 | 27     | 3.0        |
| 5X4X3 | 60     | 5.0        |
| 4X4X4 | 64     | 6.2        |
| 11X11 | 121    | 40.1       |
| 5X5X5 | 125    | 55.4       |

The curve in Fig. 5 is a plot of

$$\text{TIME} = \exp (K1 \times N) \quad (42)$$

where

K1 = an averaged constant

n = number of mesh points.

The value of  $K_1$  was arrived at by a simple, although somewhat crude, technique of averages. Five runs were made and timed, each with different numbers of mesh points. In each case the log of the time was divided by the number of mesh points. These five numbers are then averaged and yield

$$K_1 = 0.0317$$

The behavior of the implicit with respect to time required appears linear. Assuming this to be true, the governing equation is simply

$$\text{TIME} = K_2 \times N + I \quad (43)$$

where

$$I = y \text{ intercept}$$

A linear regression produced a slope of  $K_2 = 0.0472$  with an intercept of  $I = 0.4555$  (18). The data used in the plot are

TABLE VIII

Data Used in the Determination of K2.

| GRID  | POINTS | TIME (SEC) |
|-------|--------|------------|
| 3X3X3 | 27     | 1.9        |
| 5X4X3 | 60     | 3.1        |
| 4X4X4 | 64     | 3.4        |
| 11X11 | 121    | 6.1        |
| 5X5X5 | 125    | 6.5        |

The exponential time requirement is a definite handicap for the EM unless a coarse mesh is used in the original problem.

ACCURACY OF 3-D RESULTS

Little was learned concerning the accuracy of the methods except that the overall accuracy of both methods improved with the addition of more internal mesh points. This is not an unexpected result, but gratifying to know the theory correctly predicts this. Table IX lists the results of the accuracy run for the 7X7X7 grid cube.

TABLE IX

Comparison of Accuracy for N=7 Cube.

| METHOD | TBAR |      |      |      |      |      |      |      |      |      |
|--------|------|------|------|------|------|------|------|------|------|------|
|        | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    | 10   |
| EM     | 2.32 | 0.15 | 0.11 | 0.03 | 0.01 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| IM     | 3.63 | 1.79 | 0.73 | 0.21 | 0.03 | 0.10 | 0.12 | 0.11 | 0.09 | 0.07 |

Again, as in the 2-D case, the EM is more accurate than the IM. The accuracy seems to get better after TBAR greater than 5.0. The problem now has temperatures on all surfaces which, in turn, causes a rapid heating of the interior. The material will therefore reach a steady state condition faster than in the two dimensional case. This would account for the apparent exactness of the answers at late times.

NON-SYMMETRIC 3-D PROBLEM

An additional 3-D problem was run to deliberately remove any gain from symmetry. Very little was gained from the exercise, except the knowledge the method still works.

The problem was to make a parallelepiped having dimensions such that

$$a = 5$$

$$b = 4$$

$$c = 3$$

This box shape would make it very hard to take advantage of

symmetry. Since the other problems were run with full grid mode, this was a trivial problem to run.

The accuracy results were as follows

TABLE X  
Comparison of Accuracy for 5X4X3 Parallelepiped.

| METHOD | TBAR |      |      |      |      |      |      |      |      |      |
|--------|------|------|------|------|------|------|------|------|------|------|
|        | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    | 10   |
| EM     | 1.43 | 0.07 | 0.07 | 0.04 | 0.02 | 0.01 | 0.00 | 0.00 | 0.00 | 0.00 |
| IM     | 1.15 | 0.14 | 0.16 | 0.08 | 0.03 | 0.01 | 0.00 | 0.00 | 0.00 | 0.00 |

The same behavior is exhibited here as in the cube in Table IX. The answers do not appear symmetric because of the lack of symmetry in the region, as expected. The answers also appear to be more accurate in Table X than Table IX. This is due in part to less numbers of node points. The 3-D geometry caused a good increase in accuracy over the 2-D case, but the accuracy should improve a little if not as many calculations (roundoff error) are made.

## V. EVALUATION OF THE EIGENVALUE METHOD

Based upon results of the previous chapter, an overall rating is made of the eigenvalue method.

### THE STRONG POINTS

The accuracy of the EM is better than the other methods (EX,IM,CN). Even at earlier times the EM is still strong. This better accuracy remains throughout a transient, regardless of duration.

The reason given for the improved accuracy is the exactness of the time derivative in the heat equation. The other methods are similarly derived with some form of discretized approximation via Taylor series (2:347-348). The EM is even more accurate than the best of the comparison methods, CN, in the time derivative.

The appearance that the other methods approach the same accuracy of the EM at later times is misleading. Figure 4 shows the error of the comparison methods to converge to that of the EM. Mathematically, we know this to be untrue. The difference between the EM and the other methods accuracy should remain constant. This convergence is brought about by the region under consideration approaching the steady state condition.

The efficiency of the EM must lie on the side of being an asset. Consider a hypothetical problem. A small nuclear reactor (as on a submarine) has a problem requiring a

shutdown and subsequent cooling. Technical problems dictate a slow cool down taking 1 week. What temperature behavior happens 1 meter from center?

Using the EM and assuming a 1-D treatment (the reactor is a symmetric cylinder with variation only in the radial direction), how long would it take the computer to determine reactor temperatures at the point at  $t = 1$  week?

Assume each method divides the 1 meter length into 100 intervals for accuracy and allows the reference time,  $t^*$ , to be 1 second. Using Fig. (5), we find the EM requires approximately 20-25 seconds of CPU time to solve the problem. This time includes determination of the eigenvalues, eigenvectors, and coefficients. Figure 6 shows that it takes the IM approximately 5 seconds of CPU time (5 sec to reach  $\bar{\epsilon} = 10$ ). This means it requires 1/2 second to reach  $\bar{\epsilon} = 1.0$ . Since the reference time is unity,  $\bar{\epsilon} = t$  and the IM must iterate till it reaches 1 week, or 604,800 seconds. The IM would then require 302,400 CPU seconds to reach the same solution found by the EM in 20-25 seconds CPU time. There is little doubt which scheme the engineer on the submarine will choose.

Two more assets are possessed by the eigenvalue method. The first is the unconditional stability of the method brought about because of the analytical nature in the temporal domain. The second point is the omission of most of the terms in Eq. (36). This is allowed because the coefficients become so small in some cases that most terms

contribute very little to the overall answer and can be eliminated without affecting the answer. Undoubtedly, a solution requiring 5 terms runs faster than one requiring 100. This point is moot. The time saved in eliminating the extra terms is often, as in the case of large numbers of mesh points, negligible when compared to the time required to find all the eigenvalues, eigenvectors, and coefficients.

#### THE WEAK POINTS

Paradoxically, one of the EM's strongest points can also be its weakest point. It has been demonstrated the EM performs poorly with large numbers of mesh points (requires large CPU times). This may not be an insurmountable problem if one is using a very capable computer, but when restricted to a machine such as the VAX 11/780, it is a definite problem.

The heart of the problem is the requirement to find complete sets of eigenvalues, eigenvectors, and coefficients. This study used standard IMSL routines. These routines are widely available and are commonly used in such studies (20; 23). The EM has potential of being very powerful if better, faster routines were made available to find these eigenvalues/vectors. An extensive literature search (1; 3; 5; 6; 7; 11; 17), indicates this problem is being given top priority in the numerical analysis community (12).

Another weak point of the EM is the inability to hand

check any of the calculations. The nature of finding eigenvalues and eigenvectors makes for extreme difficulty in a hand check of the calculations, especially for a large matrix. Imagine trying to hand calculate a 100X100 eigenvector/value problem. This discourages the novice user from taking advantage of the EM. If the problem has no analytical solution or is not easily approximated, the EM user will have little confidence in his answers.

## VI. CONCLUSIONS AND RECOMMENDATIONS

### CONCLUSIONS

The eigenvalue method is a good numerical technique worthy of inclusion in any problem solver's collection of numerical techniques. It is not a panacea capable of solving problems other methods such as explicit, implicit, or Crank-Nicolson cannot.

Like all things, the EM has its good points and its bad points. It should be considered better in accuracy than most discretization methods. It is especially well suited in large parametric studies. It does not lend itself well to large spatial studies involving great numbers of mesh points. The improved accuracy and the slower execution times can be paired against one another to achieve a reasonably fast and accurate method.

When used as a general problem solver the EM is about the same as any other technique. It is when used in large transient time applications that the EM is vastly superior.

The EM is unconditionally stable because of the exactness of the time derivative of the temperature in the heat equation.

The EM translates easily into three dimensions. It follows the method should also work well in other coordinate systems, such as cylindrical or spherical.

There is a potential of high CPU useage time in the EM if large numbers of internal mesh points are encountered.

The EM will always be slightly better in accuracy than the standard discretization techniques used in this study. This accuracy improvement is made available by the analytical nature of the time derivative.

#### RECOMMENDATIONS

It is recommended that a more detailed investigation be made into the use of the eigenvalue method. Extensions of the method in spherical and cylindrical coordinates should be made. Also of interest would be how the EM performs on regions with irregular boundaries.

It is further recommended that a better description of the amount of CPU time required to execute the programs be given. The one expressed in this study used only five data points. More sophisticated curve fits should be employed to verify or refute the exponential nature of the time requirement of the EM.

Another recommendation is the use of the EM as presented in this study with faster, more capable machines. It would be interesting to reaccomplish this study on the HARRIS 800, CYBER, or CRAY computers.

It is recommended that a better method be determined to find eigenvalues and eigenvectors. If such methods do not yet exist, then it is suggested that careful review of the existing methods (IMSL) be made with the goal of improving their speeds and/or accuracies. Also in this area, it is recommended that a careful investigation be made to

determine if all the eigenvalues and eigenvectors need to be calculated in order to find the correct multiplying coefficients for the solution of the problem.

It is recommended that a study be made to determine a way to predict the balance between the EM accuracy and mesh point spacing. This would save much time in trial and error. Once this balance has been determined it would be instructive to make a computer study to further explore long transient problems.

It is recommended that an investigation of the method of lines (Runge-Kutta) be made and results compared to the eigenvalue method to identify similarities and differences. A time comparison between the two methods would prove extremely useful.

Finally, it is recommended that problems involving radiative surfaces, heat sources, and differing material properties (such as specific heat, thermal transmissivity, etc.) be studied using the EM. Furthermore, differing boundary conditions such as Neuman, Dirichlet, and Robins should be examined and evaluated.

## APPENDIX A

### DERIVATION OF THE ANALYTICAL SOLUTION

In this appendix, verification of the sample problem presented by Shih and Skladany will be accomplished (19:411). A three dimensional problem is also solved in this report. Therefore, it is convenient to extend the derivation to three dimensions. This was not done in the previously mentioned article.

#### THE SAMPLE PROBLEM

Given the heat equation

$$\rho c \partial T / \partial t = k \partial^2 T / \partial x^2 + k \partial^2 T / \partial y^2 + k \partial^2 T / \partial z^2 \quad (A-1)$$

with boundary conditions and initial conditions as

$$T(0, y, z, t) = T_0$$

$$T(a, y, z, t) = T_0$$

$$T(x, 0, z, t) = T_0$$

$$T(x, b, z, t) = T_0$$

$$T(x, y, 0, t) = T_0$$

$$T(x, y, c, t) = T_0$$

$$T(x, y, z, 0) = T_i$$

Now, let the normalized temperature be defined as

$$\Theta(x, y, z, t) = [T(x, y, z, t) - T_0] / (T_i - T_0) \quad (A-2)$$

Then the normalized boundary conditions are now

$$\Theta(0, y, z, t) = 0$$

$$\Theta(a, y, z, t) = 0$$

$$\Theta(x, 0, z, t) = 0$$

$$\Theta(x, b, z, t) = 0$$

$$\Theta(x, y, 0, t) = 0$$

$$\Theta(x, y, c, t) = 0$$

$$\Theta(x, y, z, 0) = 1$$

In terms of the normalized temperature, Eq. (A-1) now appears as

$$\partial\Theta/\partial t = \alpha [\partial^2\Theta/\partial x^2 + \partial^2\Theta/\partial y^2 + \partial^2\Theta/\partial z^2] \quad (A-3)$$

where

$$\alpha = k/\rho c v$$

#### SEPARATION OF VARIABLES

Assuming the method of separation of variables technique is valid (because of the same temperature on all surfaces, then there exists some solution (4:34)

$$\Theta(x, y, z, t) = X(x) Y(y) Z(z) T(t) \quad (A-4)$$

and differentiating with respect to the appropriate independent variable

$$\partial\theta/\partial t = XYZT' \quad (\text{A-5})$$

$$\partial^2\theta/\partial x^2 = X''YZT \quad (\text{A-6})$$

$$\partial^2\theta/\partial y^2 = XY''ZT \quad (\text{A-7})$$

$$\partial^2\theta/\partial z^2 = XYZ''T \quad (\text{A-8})$$

Using these, Eq. (A-3) becomes

$$XYZT' = \alpha [X''YZT + XY''ZT + XYZ''T] \quad (\text{A-9})$$

dividing both sides by XYZT

$$T'/T = \alpha [X''/X + Y''/Y + Z''/Z] \quad (\text{A-10})$$

rearranging gives the form

$$T'/kT - Y''/Y - Z''/Z = X''/X \quad (\text{A-11})$$

In typical separation of variables problems, both sides must also equal some constant, say,  $-\lambda$

$$X''/X = -\lambda \quad (\text{A-12})$$

considering the x dependence

$$X'' + \lambda X = 0 \quad (\text{A-13})$$

This equation typically has solutions of the form

$$X(x) = c_1 \sin \sqrt{\lambda} x + c_2 \cos \sqrt{\lambda} x \quad (\text{A-14})$$

applying the normalized boundary conditions

$$X(0) = c_1 \sin(0) + c_2 \cos(0) \quad (\text{A-15})$$

for this to be true  $c_2 = 0$ . Also,

$$X(a) = c_1 \sin \sqrt{\lambda} a = 0 \quad (\text{A-16})$$

to avoid the trivial solution  $c_1 \neq 0$ , therefore,

$$\sqrt{\lambda} a = n\pi \quad (\text{A-17})$$

$$\lambda = (n\pi/a)^2 \quad (\text{A-18})$$

with

$$X(x) = \sum_{n=1}^{\infty} c_n \sin(n\pi x/a) \quad (\text{A-19})$$

where

$$n = 1, 3, 5, \dots$$

Similar treatments of the y and z variables will yield

$$Y(y) = \sum_{m=1}^{\infty} C_m \sin(m\pi y/b) \quad (\text{A-20})$$

$$Z(z) = \sum_{o=1}^{\infty} C_o \sin(o\pi z/c) \quad (\text{A-21})$$

where n, m, and o are odd as from before. The time dependence is found from

$$T' + k(\lambda_n + \beta_m + \gamma_o)T = 0 \quad (\text{A-22})$$

where

$\lambda_n$  = x separation constant

$\beta_m$  = y separation constant

$\gamma_o$  = z separation constant

and let

$$k(\lambda_n + \beta_m + \gamma_o) = \{nmo$$

With these substitutions, Eq. (A-22) becomes

$$T' + \{nmo T = 0 \quad (\text{A-23})$$

whose solution has the form

$$T(t) = g \exp(-\{nmo t) \quad (\text{A-24})$$

applying the initial condition

$$T(0) = g \exp(0) = 1 \quad (\text{A-25})$$

which implies that  $g = 1$ . Therefore,

$$T(t) = \exp(-\{nmo t\}) \quad (\text{A-26})$$

Combining all the constants thus far as

$$K_{nmo} = C_n C_m C_o \quad (\text{A-27})$$

and using Eqs. (A-19), (A-20), (A-21), and (A-24) and recalling Eq. (A-3) one gets the normalized temperature solution

$$\begin{aligned} \Theta(x, y, z, t) = & \sum_{n=1}^{\infty} \sum_{m=1}^{\infty} \sum_{o=1}^{\infty} K_{nmo} \exp[-(n^2/a^2 \\ & + m^2/b^2 + o^2/c^2) k\pi^2 t] \sin(n\pi x/a) \\ & \sin(m\pi y/b) \sin(o\pi z/c) \end{aligned} \quad (\text{A-28})$$

#### FOURIER SERIES

Let  $f(x, y, z, 0) = 1$ . From Fourier series, any function may be expanded as a sine series (4:110).

$$f(x, y, z) = \sum_{m=1}^{\infty} A_n \sin(n\pi x/a) \quad (\text{A-29})$$

and

$$A_n = (2/a) \int_0^a f(x, y, z) \sin(n\pi x/a) dx \quad (\text{A-30})$$

$$A_n = (2/a) \int_0^a 1 \sin(n\pi x/a) dx \quad (\text{A-31})$$

but,  $A_n$  can also be expanded as

$$A_n = \sum_{m=1}^{\infty} B_{nm} \sin(m\pi y/b) \quad (\text{A-32})$$

where

$$B_{nm} = (2/b) \int_0^b A_n \sin(m\pi y/b) dy \quad (\text{A-33})$$

and, like above,  $B_{nm}$  can also be expanded as

$$B_{nm} = \sum_{o=1}^{\infty} X_{nmo} \sin(o\pi z/c) \quad (\text{A-34})$$

where

$$X_{nmo} = (2/c) \int_0^c B_{nm} \sin(o\pi z/c) dz \quad (\text{A-35})$$

plugging Eqs. (A-31) and (A-33) into Eq. (A-35) yields

$$K_{nmo} = (8/abc) \int_0^a \int_0^b \int_0^c \sin(n\pi x/a) \sin(m\pi y/b) \sin(o\pi z/c) dz dy dx \quad (\text{A-36})$$

The integral

$$\int_0^L \sin(n\pi \phi/L) d\phi \quad (\text{A-37})$$

has value only when  $n$  is odd and is equal to

$$2L / n\pi \quad (\text{A-38})$$

With this integral appearing three times in Eq. (A-36) the constant is clearly

$$K_{nmo} = (8/abc)(2a/n\pi)(2b/m\pi)(2c/o\pi) \quad (\text{A-39})$$

$$K_{nmo} = 64 / (nmoo\pi^3) \quad (\text{A-40})$$

### THE NORMALIZED SOLUTION

Once the value for the constant is known, it is used in Eq. (A-28) to give the result

$$\Theta(x, y, z, t) = \sum_n \sum_m \sum_o \left[ \frac{64}{(nm o \pi^3)} \right] \exp \left[ - (n^2/a^2 + m^2/b^2 + o^2/c^2) k \pi^2 t \right] \sin(n\pi x/a) \sin(m\pi y/b) \sin(o\pi z/c) \quad (A-41)$$

This is an extended version of the solution found in the Shih and Skladany article and verifies their exact solution (19:410). This solution is valid only for terms where n, m, and o are odd and not equal to zero.

APPENDIX B  
APPLICATION OF THE EIGENVALUE METHOD  
TO A SMALL SYSTEM OF EQUATIONS

The core of the eigenvalue method is the vector representation of the heat equation as

$$\{\dot{\phi}(t)\} = [D]\{\phi(t)\} \quad (\text{B-1})$$

where

$$[D] = \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{bmatrix} \quad (\text{B-2})$$

Shih and Skladany applied the above to a 2x2 matrix system (19:412). This appendix will do the same, paralleling their development with a few extra steps allowed for clarity.

Temporarily dropping the vector and matrix notation and looking only at the equations of the sample matrix, Eq. (B-1) can be written

$$\dot{\phi}(t) - D\phi(t) = 0 \quad (\text{B-3})$$

this type differential equation is recognized as having solutions of the form

$$\phi_1(t) = x_1 \exp(\lambda_1 t) + y_1 \exp(\lambda_2 t) \quad (\text{B-4})$$

$$\phi_2(t) = x_2 \exp(\lambda_1 t) + y_2 \exp(\lambda_2 t) \quad (\text{B-5})$$

differentiation with respect to time

$$\dot{\phi}_1(t) = \lambda_1 x_1 \exp(\lambda_1 t) + \lambda_2 y_1 \exp(\lambda_2 t) \quad (\text{B-6})$$

$$\dot{\phi}_2(t) = \lambda_1 x_2 \exp(\lambda_1 t) + \lambda_2 y_2 \exp(\lambda_2 t) \quad (\text{B-7})$$

plugging Eqs. (B-6) and (B-7) into (B-4) and (B-5) gets

$$\begin{aligned} \lambda_1 x_1 \exp(\lambda_1 t) + \lambda_2 y_1 \exp(\lambda_2 t) = d_{11} [(x_1 \exp(\lambda_1 t) \\ + y_1 \exp(\lambda_2 t))] + d_{12} [(x_2 \exp(\lambda_1 t) + y_2 \exp(\lambda_2 t))] \end{aligned} \quad (\text{B-8})$$

$$\begin{aligned} \lambda_1 x_2 \exp(\lambda_1 t) + \lambda_2 y_2 \exp(\lambda_2 t) = d_{21} [(x_1 \exp(\lambda_1 t) \\ + y_1 \exp(\lambda_2 t))] + d_{22} [(x_2 \exp(\lambda_1 t) + y_2 \exp(\lambda_2 t))] \end{aligned} \quad (\text{B-9})$$

placing everything on the right side and simplifying

$$0 = \exp(\lambda_1 t) (d_{11} x_1 + d_{12} x_2 - \lambda_1 x_1) + \exp(\lambda_2 t) (d_{11} y_1 + d_{12} y_2 - \lambda_2 y_1) \quad (\text{B-10})$$

$$0 = \exp(\lambda_1 t) (d_{21} x_1 + d_{22} x_2 - \lambda_1 x_2) + \exp(\lambda_2 t) (d_{21} y_1 + d_{22} y_2 - \lambda_2 y_2) \quad (\text{B-11})$$

Equations (B-10) and (B-11) when written in matrix form are

$$0 = \exp(\lambda_1 t) \begin{bmatrix} d_{11} - \lambda_1 & d_{12} \\ d_{21} & d_{22} - \lambda_1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \exp(\lambda_2 t) \begin{bmatrix} d_{11} - \lambda_2 & d_{12} \\ d_{21} & d_{22} - \lambda_2 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \quad (\text{B-12})$$

from Eq. (B-2)

$$[D] = \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{bmatrix}$$

which implies  $\lambda_1$  and  $\lambda_2$  are the eigenvalues of  $[D]$  and  $[x_1, x_2]^T$  and  $[y_1, y_2]^T$  are the corresponding eigenvectors. The terms  $\exp(\lambda_1 t)$  and  $\exp(\lambda_2 t)$  are just multiplying scalars. If  $\lambda_1, \lambda_2, [x_1, x_2]^T$ , and  $[y_1, y_2]^T$  can be determined, Eq. (B-1) is solved.

## APPENDIX C

### STANDARD IMSL ROUTINES USED IN THE STUDY

This appendix contains copies of the standard IMSL routines used in the various programs in the study. This appendix is provided to the reader to aide in the evaluation of the solving routines used. Any questions concerning method of solution once the programs called these routines should be refered to the subroutine author.

IMSL ROUTINE NAME - EIGRF

PURPOSE - EIGENVALUES AND (OPTIONALLY) EIGENVECTORS OF A REAL GENERAL MATRIX IN FULL STORAGE MODE

USAGE - CALL EIGRF (A,N,IA,IJOB,W,Z,IZ,WK,IER)

ARGUMENTS

A - THE INPUT REAL GENERAL MATRIX OF ORDER N WHOSE EIGENVALUES AND EIGENVECTORS ARE TO BE COMPUTED. INPUT A IS DESTROYED IF IJOB IS EQUAL TO 0 OR 1.

N - THE INPUT ORDER OF THE MATRIX A.

IA - THE INPUT ROW DIMENSION OF MATRIX A EXACTLY AS SPECIFIED IN THE DIMENSION STATEMENT IN THE CALLING PROGRAM.

IJOB - THE INPUT OPTION PARAMETER. WHEN  
 IJOB = 0, COMPUTE EIGENVALUES ONLY  
 IJOB = 1, COMPUTE EIGENVALUES AND EIGENVECTORS.  
 IJOB = 2, COMPUTE EIGENVALUES, EIGENVECTORS AND PERFORMANCE INDEX.  
 IJOB = 3, COMPUTE PERFORMANCE INDEX ONLY. IF THE PERFORMANCE INDEX IS COMPUTED, IT IS RETURNED IN WK(1). THE ROUTINES HAVE PERFORMED (WELL, SATISFACTORILY, POORLY) IF WK(1) IS (LESS THAN 1, BETWEEN 1 AND 100, GREATER THAN 100).

W - THE OUTPUT COMPLEX VECTOR OF LENGTH N, CONTAINING THE EIGENVALUES OF A.  
 NOTE - THE ROUTINE TREATS W AS A REAL VECTOR OF LENGTH 2\*N. AN APPROPRIATE EQUIVALENCE STATEMENT MAY BE REQUIRED. SEE DOCUMENT EXAMPLE.

Z - THE OUTPUT N BY N COMPLEX MATRIX CONTAINING THE EIGENVECTORS OF A.  
 THE EIGENVECTOR IN COLUMN J OF Z CORRESPONDS TO THE EIGENVALUE W(J).  
 IF IJOB = 0, Z IS NOT USED.  
 NOTE - THE ROUTINE TREATS Z AS A REAL VECTOR OF LENGTH 2\*N\*N. AN APPROPRIATE EQUIVALENCE STATEMENT MAY BE REQUIRED. SEE DOCUMENT EXAMPLE.

IZ - THE INPUT ROW DIMENSION OF MATRIX Z EXACTLY AS SPECIFIED IN THE DIMENSION STATEMENT IN THE CALLING PROGRAM. IZ MUST BE GREATER THAN OR EQUAL TO N IF IJOB IS NOT EQUAL TO ZERO.

WK - WORK AREA, THE LENGTH OF WK DEPENDS ON THE VALUE OF IJOB, WHEN  
 IJOB = 0, THE LENGTH OF WK IS AT LEAST N.  
 IJOB = 1, THE LENGTH OF WK IS AT LEAST 2N.  
 IJOB = 2, THE LENGTH OF WK IS AT LEAST (2+N)N.  
 IJOB = 3, THE LENGTH OF WK IS AT LEAST 1.

IER - ERROR PARAMETER. (OUTPUT)  
 TERMINAL ERROR  
 IER = 128+J, INDICATES THAT EQRH3F FAILED TO CONVERGE ON EIGENVALUE J. EIGENVALUES J+1, J+2, ..., N HAVE BEEN COMPUTED CORRECTLY. EIGENVALUES 1, ..., J ARE SET TO ZERO.  
 IF IJOB = 1 OR 2 EIGENVECTORS ARE SET TO ZERO. THE PERFORMANCE INDEX IS SET TO 1000.  
 WARNING ERROR (WITH FIX)  
 IER = 66, INDICATES IJOB IS LESS THAN 0 OR IJOB IS GREATER THAN 3. IJOB SET TO 1.  
 IER = 67, INDICATES IJOB IS NOT EQUAL TO ZERO, AND IZ IS LESS THAN THE ORDER OF MATRIX A. IJOB IS SET TO ZERO.

PRECISION/HARDWARE - SINGLE AND DOUBLE/H32  
 - SINGLE/H36, H48, H60

REQD. IMSL ROUTINES - EBALAF, EBBCKF, EHBCKF, EHESF, EQRH3F, UERTST, UGETIO

NOTATION - INFORMATION ON SPECIAL NOTATION AND CONVENTIONS IS AVAILABLE IN THE MANUAL INTRODUCTION OR THROUGH IMSL ROUTINE UHELP

### Algorithm

EIGRF computes eigenvalues and (optionally) eigenvectors of a real matrix. It can also compute a performance index.

EIGRF calls IMSL routine EBALAF to balance the matrix. Then, EHESF and EQRH3F are called to compute eigenvalues and (optionally) eigenvectors. When eigenvectors are computed, EHBCKF and EBBCKF are called to backtransform the eigenvectors.

The performance index is defined as follows

$$P = \max_{1 \leq j \leq n} \frac{\|Az^j - w_j z^j\|_1}{\|A\|_1 \|z^j\|_1} 10(N) \text{ (EPS)}$$

where the max is taken over the j eigenvalues  $w_j$  and associated eigenvectors  $z^j$ . EPS specifies the relative precision of floating point arithmetic. When P is less than 1, the performance of the routines is considered to be excellent in the sense that the residuals  $Az - wz$  are as small as can be expected. When P is between 1 and 100 the performance is good. When P is greater than 100 the performance is considered poor.

The performance index was first developed and used by the EISPACK project at Argonne National Laboratory.

See references:

1. Wilkinson, J. H., The Algebraic Eigenvalue Problem, Clarendon Press, Oxford, 1965.
2. Smith, B. T., Boyle, J. M., Garbow, B. S., Ikebe, Y., Klema, V. C., and Moler, C. B., Matrix Eigensystem Routines, Springer-Verlag, 1974.

Programming Notes

1. A is preserved when IJOB=2 or 3 and N=IA. When N<IA, rows N+1, N+2, ..., IA of A are destroyed. In all other cases A is destroyed.
2. The eigenvalues are unordered except that complex conjugate pairs of eigenvalues appear consecutively with the eigenvalue having the positive imaginary part first.
3. The eigenvectors are not normalized.
4. When IJOB=3 (i.e., to compute a performance index only) the eigenvalues, W, and eigenvectors, Z, are assumed to be input.

Example 1

In this example, EIGRF is called to compute eigenvalues, eigenvectors and a performance index by setting input IJOB=2. After the call to EIGRF, the eigenvectors are normalized. For machines which require equivalencing, see example 2.

```
INTEGER  N,IA,IJOB,IZ,IER
REAL     A(4,4),WK(24)
COMPLEX  W(4),Z(4,4),ZN
```

Input:

```
IA   = 4
IZ   = 4
N    = 4
IJOB = 2
```

```
A   =  $\begin{bmatrix} 4.0 & -5.0 & 0.0 & 3.0 \\ 0.0 & 4.0 & -3.0 & -5.0 \\ 5.0 & -3.0 & 4.0 & 0.0 \\ 3.0 & 0.0 & 5.0 & 4.0 \end{bmatrix}$ 
```

```
CALL EIGRF (A,N,IA,IJOB,W,Z,IZ,WK,IER)
```

NORMALIZE EIGENVECTORS

```
DO 5 J=1,N
  ZN = Z(1,J)
  DO 5 I=1,N
    Z(I,J) = Z(I,J)/ZN
```

```
CONTINUE
```

Output:

IER = 0

W = (12.0, 1.0+5.0i, 1.0-5.0i, 2.0) (eigenvalues)

Z =  $\begin{bmatrix} 1.0 \\ -1.0 \\ 1.0 \\ 1.0 \end{bmatrix}, \begin{bmatrix} 1.0 \\ 0.0 \\ 0.0 \\ -1.0 \end{bmatrix} + i \begin{bmatrix} 0.0 \\ -1.0 \\ -1.0 \\ 0.0 \end{bmatrix}, \begin{bmatrix} 1.0 \\ 0.0 \\ 0.0 \\ -1.0 \end{bmatrix} + i \begin{bmatrix} 0.0 \\ 1.0 \\ 1.0 \\ 0.0 \end{bmatrix}, \begin{bmatrix} 1.0 \\ 1.0 \\ -1.0 \\ 1.0 \end{bmatrix}$  (eigenvectors)

WK(1) < 1.0 (performance index)

Example 2

For machines which require an equivalence statement in situations where an array is of one type in the calling program but of another type in the subroutine, EIGRF should be called as follows.

```
INTEGER      N, IA, IJOB, IZ, IER
REAL         A(4,4), WK(24), RW(8), RZ(32)
COMPLEX      W(4), Z(4,4), ZN
EQUIVALENCE (W(1), RW(1)), (Z(1,1), RZ(1))
```

Input:

```
IA = 4
IZ = 4
N = 4
IJOB = 2
```

A =  $\begin{bmatrix} 4.0 & -5.0 & 0.0 & 3.0 \\ 0.0 & 4.0 & -3.0 & -5.0 \\ 5.0 & -3.0 & 4.0 & 0.0 \\ 3.0 & 0.0 & 5.0 & 4.0 \end{bmatrix}$

```
CALL EIGRF (A,N,IA,IJOB,RW,RZ,IZ,WK,IER)
C                                     NORMALIZE EIGENVECTORS
DO 5 J=1,N
  ZN = Z(1,J)
  DO 5 I=1,N
    Z(I,J) = Z(I,J)/ZN
5 CONTINUE
```

Output:

IER = 0

W = (12.0, 1.0+5.0i, 1.0-5.0i, 2.0) (eigenvalues)

Z =  $\begin{bmatrix} 1.0 \\ -1.0 \\ 1.0 \\ 1.0 \end{bmatrix}, \begin{bmatrix} 1.0 \\ 0.0 \\ 0.0 \\ -1.0 \end{bmatrix} + i \begin{bmatrix} 0.0 \\ -1.0 \\ -1.0 \\ 0.0 \end{bmatrix}, \begin{bmatrix} 1.0 \\ 0.0 \\ 0.0 \\ -1.0 \end{bmatrix} + i \begin{bmatrix} 0.0 \\ 1.0 \\ 1.0 \\ 0.0 \end{bmatrix}, \begin{bmatrix} 1.0 \\ 1.0 \\ -1.0 \\ 1.0 \end{bmatrix}$  (eigenvectors)

WK(1) < 1.0 (performance index)

IMSL ROUTINE NAME - LEQT2F

PURPOSE - LINEAR EQUATION SOLUTION - FULL STORAGE  
MODE - HIGH ACCURACY SOLUTION

USAGE - CALL LEQT2F (A,M,N,IA,B,IDGT,WKAREA,IER)

ARGUMENTS

A - INPUT MATRIX OF DIMENSION N BY N CONTAINING  
THE COEFFICIENT MATRIX OF THE EQUATION  
AX = B.

M - NUMBER OF RIGHT-HAND SIDES. (INPUT)

N - ORDER OF A AND NUMBER OF ROWS IN B. (INPUT)

IA - ROW DIMENSION OF A AND B EXACTLY AS SPECIFIED  
IN THE DIMENSION STATEMENT IN THE CALLING  
PROGRAM. (INPUT)

B - INPUT MATRIX OF DIMENSION N BY M CONTAINING  
THE RIGHT-HAND SIDES OF THE EQUATION AX = B.  
ON OUTPUT, THE N BY M MATRIX OF SOLUTIONS  
REPLACES B.

IDGT - INPUT OPTION.  
IF IDGT IS GREATER THAN 0, THE ELEMENTS OF  
A AND B ARE ASSUMED TO BE CORRECT TO IDGT  
DECIMAL DIGITS AND THE ROUTINE PERFORMS  
AN ACCURACY TEST.  
IF IDGT EQUALS 0, THE ACCURACY TEST IS  
BYPASSED.  
ON OUTPUT, IDGT CONTAINS THE APPROXIMATE  
NUMBER OF DIGITS IN THE ANSWER WHICH  
WERE UNCHANGED AFTER IMPROVEMENT.

WKAREA - WORK AREA OF DIMENSION GREATER THAN OR EQUAL  
TO  $N^2+3N$ .

IER - ERROR PARAMETER. (OUTPUT)  
WARNING ERROR  
IER = 34 INDICATES THAT THE ACCURACY TEST  
FAILED. THE COMPUTED SOLUTION MAY BE IN  
ERROR BY MORE THAN CAN BE ACCOUNTED FOR  
BY THE UNCERTAINTY OF THE DATA. THIS  
WARNING CAN BE PRODUCED ONLY IF IDGT IS  
GREATER THAN 0 ON INPUT. (SEE THE  
CHAPTER L PRELUDE FOR FURTHER DISCUSSION.)  
TERMINAL ERROR  
IER = 129 INDICATES THAT THE MATRIX IS  
ALGORITHMICALLY SINGULAR. (SEE THE  
CHAPTER L PRELUDE).  
IER = 131 INDICATES THAT THE MATRIX IS TOO  
ILL-CONDITIONED FOR ITERATIVE IMPROVEMENT  
TO BE EFFECTIVE.

PRECISION/HARDWARE - SINGLE AND DOUBLE/H32  
- SINGLE/H36,H48,H60

REQD. IMSL ROUTINES - SINGLE/LUDATN,LUELMN,LUREFN,UERTST,UGETIO  
- DOUBLE/LUDATN,LUELMN,LUREFN,UERTST,UGETIO,  
VXADD,VXMUL,VXSTO

NOTATION - INFORMATION ON SPECIAL NOTATION AND  
CONVENTIONS IS AVAILABLE IN THE MANUAL  
INTRODUCTION OR THROUGH IMSL ROUTINE UHELP

## Algorithm

LEQT2F solves the set of linear equations  $AX=B$  for  $X$ , where  $A$  is the  $N$  by  $N$  matrix and is in full storage mode.  $B$  is  $N$  by  $M$ . The difference between this routine and routine LEQT1F is that LEQT2F invokes iterative improvement if necessary, in order to improve the accuracy of the solution  $X$ .

The routine performs Gaussian elimination (Crout algorithm) with equilibration, partial pivoting, and iterative improvement as required.

See reference:

Forsythe, George and Moler, Cleve B., Computer Solution of Linear Algebraic Systems, Englewood Cliffs, N. J., Prentice-Hall, Inc., 1967, Chapters 9, 13, 24.

## Programming Notes

1. Iterative improvement is costly in both computer time and storage. When high accuracy is not needed, subroutine LEQT1F may be used to advantage.
2. When  $IA$  is greater than  $N$ , elements of  $A$  in rows  $N+1$  to  $IA$  are used as workspace and are destroyed. However, the first  $N$  rows of  $A$  are restored to their original content on exit from LEQT2F.

## Accuracy

If  $IDGT$  is greater than zero, elements of  $A$  are assumed to be correct to  $IDGT$  decimal digits. The solution  $X$  will be the exact solution, without any roundoff error, to a matrix  $\bar{A}$  whose elements agree with the elements of  $A$  in the first  $IDGT$  decimal digits. The program first attempts such a solution without iterative improvement. Then iterative improvement is performed if necessary. If this also fails, solution is not possible and the program exits. Upon exit, the first columns of  $B$  will have been replaced by the best solution that the computer can generate and  $IDGT$  is set to the approximate number of digits in the answer which were unchanged by the improvement (see IMSL routine LUREFF). The other columns of  $B$  are left unchanged in this case and  $IER$  is set to 131. If input  $IDGT$  equals zero, iterative improvement is automatically performed.

## Example

This example inputs the 3 by 3 matrix  $A$  and the 3 by 4 matrix  $B$  solving for the 3 by 4 matrix  $X$  of  $AX=B$ .  $X$  overwrites  $B$  on output.

Input:

```
REAL A(4,4),B(4,4),WKAREA(18)
INTEGER M,N,IA,IDGT,IER
N      = 3
M      = 4
IA     = 4
IDGT   = 3
```

$$A = \begin{bmatrix} 33.000 & 16.0 & 72.0 & x \\ -24.000 & -10.0 & -57.0 & x \\ -8.000 & -4.0 & -17.0 & x \\ x & x & x & x \end{bmatrix}$$

$$B = \begin{bmatrix} 1.0 & 0.0 & 0.0 & -359.0 \\ 0.0 & 1.0 & 0.0 & 281.0 \\ 0.0 & 0.0 & 1.0 & 85.0 \\ x & x & x & x \end{bmatrix}$$

CALL LEQT2F(A,M,N,IA,B, IDGT,WKAREA, IER)

Output:

IDGT = 3  
IER = 0

$$B = \begin{bmatrix} -9.66666 & -2.66667 & -32. & 1. \\ 8.0 & 2.5 & 25.5 & -2. \\ 2.66667 & .666667 & 9. & -5. \\ x & x & x & x \end{bmatrix}$$

Note: x indicates elements not used by LEQT2F.

IMSL ROUTINE NAME - EIGRS

PURPOSE - EIGENVALUES AND (OPTIONALLY) EIGENVECTORS OF A REAL SYMMETRIC MATRIX

USAGE - CALL EIGRS (A,N,JOBN,D,Z,IZ,WK,IER)

ARGUMENTS

A - INPUT REAL SYMMETRIC MATRIX OF ORDER N, WHOSE EIGENVALUES AND EIGENVECTORS ARE TO BE COMPUTED. INPUT A IS DESTROYED IF IJOB IS EQUAL TO 0 OR 1.

N - INPUT ORDER OF THE MATRIX A.

JOBN - INPUT OPTION PARAMETER. IF JOBN.GE.10 A IS ASSUMED TO BE IN FULL STORAGE MODE (IN THIS CASE, A MUST BE DIMENSIONED EXACTLY N BY N IN THE CALLING PROGRAM). IF JOBN.LT.10 THEN A IS ASSUMED TO BE IN SYMMETRIC STORAGE MODE. DEFINE IJOB=MOD(JOBN,10). THEN WHEN  
 IJOB = 0, COMPUTE EIGENVALUES ONLY  
 IJOB = 1, COMPUTE EIGENVALUES AND EIGENVECTORS.  
 IJOB = 2, COMPUTE EIGENVALUES, EIGENVECTORS AND PERFORMANCE INDEX.  
 IJOB = 3, COMPUTE PERFORMANCE INDEX ONLY. IF THE PERFORMANCE INDEX IS COMPUTED, IT IS RETURNED IN WK(1). THE ROUTINES HAVE PERFORMED (WELL, SATISFACTORILY, POORLY) IF WK(1) IS (LESS THAN 1, BETWEEN 1 AND 100, GREATER THAN 100).

D - OUTPUT VECTOR OF LENGTH N, CONTAINING THE EIGENVALUES OF A IN ASCENDING ORDER.

Z - OUTPUT N BY N MATRIX CONTAINING THE EIGENVECTORS OF A. THE EIGENVECTOR IN COLUMN J OF Z CORRESPONDS TO THE EIGENVALUE D(J). IF IJOB = 0, Z IS NOT USED.

IZ - INPUT ROW DIMENSION OF MATRIX Z EXACTLY AS SPECIFIED IN THE DIMENSION STATEMENT IN THE CALLING PROGRAM.

WK - WORK AREA, THE LENGTH OF WK DEPENDS ON THE VALUE OF IJOB, WHEN  
 IJOB = 0, THE LENGTH OF WK IS AT LEAST N.  
 IJOB = 1, THE LENGTH OF WK IS AT LEAST N.  
 IJOB = 2, THE LENGTH OF WK IS AT LEAST  $N(N+1)/2+N$ .  
 IJOB = 3, THE LENGTH OF WK IS AT LEAST 1.

IER - ERROR PARAMETER (OUTPUT)  
 TERMINAL ERROR  
 IER = 128+J, INDICATES THAT EQRT2S FAILED TO CONVERGE ON EIGENVALUE J. EIGENVALUES AND EIGENVECTORS 1,...,J-1 HAVE BEEN COMPUTED CORRECTLY, BUT THE EIGENVALUES ARE UNORDERED. THE PERFORMANCE INDEX IS SET TO 1000.0

WARNING ERROR (WITH FIX)

IN THE FOLLOWING, IJOB = MOD(JOBN,10).  
IER = 66, INDICATES IJOB IS LESS THAN 0 OR  
IJOB IS GREATER THAN 3. IJOB SET TO 1.  
IER = 67, INDICATES IJOB IS NOT EQUAL TO  
ZERO, AND IZ IS LESS THAN THE ORDER OF  
MATRIX A. IJOB IS SET TO ZERO.

PRECISION/HARDWARE - SINGLE AND DOUBLE/H32  
- SINGLE/H36,H48,H60

REQD. IMSL ROUTINES - EHOBKS,EHOUSS,EQRT2S,UERTST,UGETIO

NOTATION - INFORMATION ON SPECIAL NOTATION AND  
CONVENTIONS IS AVAILABLE IN THE MANUAL  
INTRODUCTION OR THROUGH IMSL ROUTINE UHELP

Algorithm

EIGRS calls IMSL routine EHOUSS and EQRT2S to compute eigenvalues and (optionally) eigenvectors. When eigenvectors are computed, EHOBKS is called to backtransform the eigenvectors.

The performance index is defined as follows:

$$P = \max_{1 \leq j \leq n} \frac{\|Az^j - d_j z^j\|_1}{\|A\|_1 \|z^j\|_1 10(N)(EPS)}$$

where the max is taken over the n eigenvalues  $d_j$  and associated eigenvectors  $z^j$ . EPS specifies the relative precision of floating point arithmetic. When P is less than 1, the performance of the routines is considered to be excellent in the sense that the residuals  $Az - dz$  are as small as can be expected. When P is between 1 and 100 the performance is good. When P is greater than 100 the performance is considered poor.

The performance index was first developed and used by the EISPACK project at Argonne National Laboratory.

See references:

1. Wilkinson, J. H., The Algebraic Eigenvalue Problem, Clarendon Press, Oxford, 1965.
2. Smith, B. T., Boyle, J. M., Garbow, B. S., Ikebe, Y., Klema, V. C., and Moler, C. B., Matrix Eigensystem Routines, Springer-Verlag, 1974.

Programming Notes

1. A is preserved when IJOB=2 or 3 (IJOB=MOD(JOBN,10)). In all other cases A is destroyed.
2. The computed eigenvectors are normalized to each have Euclidean length 1.

Input:

INTEGER N,JOBN,IZ,IER  
REAL A(10),D(4),Z(4,4),WK(14)  
N = 4  
IZ = 4

A =  $\begin{bmatrix} 5.0 & & & \\ 4.0 & 5.0 & & \\ 1.0 & 1.0 & 4.0 & \\ 1.0 & 1.0 & 2.0 & 4.0 \end{bmatrix}$  In symmetric storage mode A(1)=5.0,  
A(2)=4.0,A(3)=5.0,...,A(10)=4.0

JOBN = 2  
CALL EIGRS (A,N,JOBN,D,Z,IZ,WK,IER)

Output:

IER = 0  
D = (1.0, 2.0, 5.0, 10.0) (eigenvalues)

Z =  $\begin{bmatrix} -0.70710 & 0.00000 & -0.31622 & 0.63245 \\ 0.70710 & 0.00000 & -0.31622 & 0.63245 \\ 0.00000 & 0.70710 & 0.63245 & 0.31622 \\ 0.00000 & -0.70710 & 0.63245 & 0.31622 \end{bmatrix}$  (eigenvectors)

WK(1) < 1 (performance index)

Note: Z is unique to within a sign change for each column.

## Bibliography

1. Ames, W. F. Numerical Methods for Partial Differential Equations. New York: Academic Press, 1977.
2. Beyer, W. H. CRC Standard Mathematical Tables, 26th Edition. Boca Raton, FL: CRC Press, Inc., 1981.
3. Carslaw, H. S. and J. C. Jaeger. Conduction of Heat in Solids. London: Oxford University Press, 1959.
4. Churchill, R. V. and J. W. Brown. Fourier Series and Boundary Value Problems, 3rd Edition. New York: McGraw-Hill Book Co., 1963.
5. Clark, M. and K. F. Hansen. Numerical Methods of Reactor Analysis. New York: Academic Press, 1964.
6. Conte, S. D. Elementary Numerical Analysis, 2nd Edition. New York: McGraw-Hill Book Co., 1972.
7. Crandall, S. H. Engineering Analysis. New York: McGraw-Hill Book Co., 1956.
8. Crank J. and P. Nicolson. "A Practical Method for Numerical Evaluations of Solutions of Partial Differential Equations of the Heat-Conduction Type," Proceedings of the Cambridge Philosophical Society, Vol. 43: 50-67 (1947).
9. Dusenberre, G. M. Numerical Analysis of Heat Flow. New York: McGraw-Hill Book Co., 1949.
10. IMSL LIB-0009. IMSL LIBRARY, Reference Manual, Edition 9; Computer library for FORTRAN subroutines. Houston, IMSL Inc., 1982.
11. Johnson, L. W. and R. D. Riess. Numerical Analysis, 2nd Edition. Reading, Mass: Addison-Wesley Publishing Co., 1982.
12. Jones, Dr. J. Professor, Mathematics. Personal interview. School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH, 11 December 1984.
13. Kaplan, Dr. B. Professor, Nuclear Physics. Personal interview. School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH, 11 December 1984.
14. Kreyszig, E. Advanced Engineering Mathematics, Fifth Edition. New York: John Wiley and Sons, 1983.

15. Kropf, Capt. R. F. Oscillatory Solutions of the Peaceman Rachford Alternating Direction Implicit Method and a Comparison of Finite Difference Methods for the Solution of the Two Dimensional Heat Diffusion Equation. MS thesis. School of Engineering. Air Force Institute of Technology (AU). Wright-Patterson AFB, OH, January, 1985.
16. McGilton, H. and R. Morgan. Introducing the Unix System. New York: McGraw-Hill Book Co., 1983.
17. Patankar, S. V. Numerical Heat Transfer and Fluid Flow. New York: McGraw-Hill Book Co., 1980.
18. Personal Programing: TI 59 programable calculator manual. Dallas, Texas Instruments Inc., 1979.
19. Shih, T. M. and J. T. Skladany. "An Eigenvalue Method for Solving Transient Heat Conduction Problems," Numerical Heat Transfer, Vol. 6: 409-422 (1983).
20. Smith, B. T., J. M. Boyle, B. S. Garbow, Y. Ikebe, V. C. Klema, and C. B. Moler. Matrix Eigensystem Routines. Springer-Verlag, Berlin: 1974.
21. Smith, G. D. Numerical Solution of Partial Differential Equations. London: Oxford University Press, 1965.
22. Thibault, J. "A Three-Dimensional Numerical Method Based on the Superposition Principle," Numerical Heat Transfer, Vol 7: 127-145 (1984).
23. Towler, B. F. and R. Y. K. Yang. "On Comparing the Accuracy of Some Finite-Difference Methods for Parabolic Partial Differential Equations," International Journal for Numerical Methods in Engineering, Vol. 14: 1021-1035 (1979).
24. Varga, R. S. Matrix Iterative Analysis. Englewood Cliffs, NJ: Prentice Hall, Inc., 1962.

## VITA

Captain David W. Landry was born on 7 October 1954 in Winshire, Texas. He graduated from Samuel Clemens High School, San Antonio, Texas, in 1972. He attended Texas A&M University in College Station, Texas, from which he recieved the degree of Bachelor of Science in Nuclear Engineering in August of 1976. Upon graduation, he recieved a commission in the USAF through the ROTC program and was called to active duty in October of 1976. He completed navigator and electronic warfare training at Mather AFB, Sacramento, California, and recieved his wings in March of 1978. He then served as an Electronic Warfare Officer on a B-52 crew at Carswell AFB, Fort Worth, Texas, until entering the School of Engineering, Air Force Institute of Technology, in October of 1983.

Permanent address: 5206 Plaza Dr.  
Schertz, Tx 78154

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

|   |   |   |                      |
|---|---|---|----------------------|
| 1a. REPORT SECURITY CLASSIFICATION<br><b>UNCLASSIFIED</b>   |   | 1b. RESTRICTIVE MARKINGS  |                      |
| 2a. SECURITY CLASSIFICATION AUTHORITY   |   | 3. DISTRIBUTION/AVAILABILITY OF REPORT<br>Approved for public release;<br>distribution unlimited. |                      |
| 2b. DECLASSIFICATION/DOWNGRADING SCHEDULE   |   | 4. PERFORMING ORGANIZATION REPORT NUMBER(S)<br>AFIT/GNE/ENP/85M -12                               |                      |
| 4. PERFORMING ORGANIZATION REPORT NUMBER(S)<br>AFIT/GNE/ENP/85M -12   |   | 5. MONITORING ORGANIZATION REPORT NUMBER(S)   |                      |
| 6a. NAME OF PERFORMING ORGANIZATION<br>School of Engineering  | 6b. OFFICE SYMBOL<br>(If applicable)<br>AFIT/ENP                  | 7a. NAME OF MONITORING ORGANIZATION   |                      |
| 6c. ADDRESS (City, State and ZIP Code)<br>Air Force Institute of Technology<br>Wright-Patterson AFB, Ohio 45433   |   | 7b. ADDRESS (City, State and ZIP Code)  |                      |
| 8a. NAME OF FUNDING/SPONSORING ORGANIZATION   | 8b. OFFICE SYMBOL<br>(If applicable)                              | 9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER   |                      |
| 8c. ADDRESS (City, State and ZIP Code)  |   | 10. SOURCE OF FUNDING NOS.  |                      |
|   |   | PROGRAM ELEMENT NO.   | PROJECT NO.          |
|   |   | TASK NO.  | WORK UNIT NO.        |
| 11. TITLE (Include Security Classification)<br>See Box 19   |   |   |                      |
| 12. PERSONAL AUTHOR(S)<br>David W. Landry, B.S., Captain, USAF  |   |   |                      |
| 13a. TYPE OF REPORT<br>MS Thesis  | 13b. TIME COVERED<br>FROM _____ TO _____                          | 14. DATE OF REPORT (Yr., Mo., Day)<br>1985 January  | 15. PAGE COUNT<br>88 |
| 16. SUPPLEMENTARY NOTATION  |   |   |                      |
| 17. COSATI CODES  |   | 18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)                 |                      |
| FIELD   | GROUP   | SUB. GR.  |                      |
| 12  | 01  | Numerical Analysis, Numerical Methods, Heat Transfer, and Eigenvalue Method                       |                      |
| 19. ABSTRACT (Continue on reverse if necessary and identify by block number)  |   |   |                      |
| Title: EVALUATION OF THE EIGENVALUE METHOD IN THE SOLUTION OF TRANSIENT HEAT CONDUCTION PROBLEMS  |   |   |                      |
| Thesis Chairman: Dr. Bernard Kaplan   |   |   |                      |
| Approved for public release: IAW AFB 100-17<br>Lynn E. Wolaver<br>Dean for Research and Professional Development<br>Air Force Institute of Technology (AFIT)<br>Wright-Patterson AFB OH 45433 |   |   |                      |
| 20. DISTRIBUTION/AVAILABILITY OF ABSTRACT<br>UNCLASSIFIED/UNLIMITED <input checked="" type="checkbox"/> SAME AS RPT. <input type="checkbox"/> DTIC USERS <input type="checkbox"/>             |   | 21. ABSTRACT SECURITY CLASSIFICATION<br>UNCLASSIFIED  |                      |
| 22a. NAME OF RESPONSIBLE INDIVIDUAL<br>Dr. Bernard Kaplan   | 22b. TELEPHONE NUMBER<br>(Include Area Code)<br>513-255-2012/4877 | 22c. OFFICE SYMBOL<br>AFIT/ENP  |                      |

The eigenvalue method, as presented by T. M. Shih and J. T. Skladany, is evaluated to determine the advantages and disadvantages of the method as compared to fully explicit, fully implicit, and Crank-Nicolson methods. Time comparisons and accuracy comparisons are made in an effort to rank the eigenvalue method in relation to the comparison schemes. → The purpose of this thesis is to verify

Shih and Skladany's original results are verified by duplicating their efforts with the method. The eigenvalue method is used to solve the parabolic heat equation in multidimensions with transient temperatures. Extensions into three dimensions are made to determine the method's feasibility in handling large geometry problems requiring great numbers of internal mesh points.

The eigenvalue method proves to be slightly better in accuracy than the comparison routines because of an exact treatment, as opposed to a numerical approximation, of the time derivative in the heat equation. It is an unconditionally stable method. It has the potential of being a very powerful routine in solving long transient type problems. The method is not well suited to finely meshed grid arrays or large regions because of the time and memory requirements necessary for calculating large sets of eigenvalues and eigenvectors. keywords:

**END**

**FILMED**

**6-85**

**DTIC**