A COMPUTER PROGRAM FOR THE ANALYSIS OF TWO-DIMENSIONAL HEAT CONDUCTION USING THE FINITE ELEMENT TECHNIQUE

by

Allan Bischof Chaloupka

Errata to the NPS Thesis entitled

A COMPUTER PROGRAM FOR THE ANALYSIS OF TWO-DIMENSIONAL HEAT CONDUCTION USING THE FINITE ELEMENT TECHNIQUE

by Allan Bischof Chaloupka

June 1969

In the main program listing:

page 72 Change card number 155 to read

```
DO 601 N=1, NUMNP                                          155
```

page 73 Remove card preceding card number 180 that reads

```
189   CALL SYMSOL (1)
```

page 74 Change statement number on card number 203 to read

```
1020 FORMAT (6I5,D10.3)                                    203
```

In SUBROUTINE FORM:

page 76 Insert a missing card between card numbers 285 and 287 to read

```
CV = SPHT(MTYPE) * CFUNC(MTYPE,TMEAN)                      286
```

In the MAIN Program listing:

page 72 Change Card Number 138 to read

```
TEMP=TEMPB(N) - TØ                                         138
```

Change the + signs to - signs on the following Card Numbers:

144 } (Note: + Q is OUT)
145 }
146
147
149
151

TABLE OF CONTENTS

ABSTRACT - 2

TABLE OF CONTENTS - 3

LIST OF FIGURES - 7

LIST OF SYMBOLS - 9

ACKNOWLEDGEMENTS - 10

I. INTRODUCTION - 11

A. REASONS FOR THE FINITE ELEMENT ANALYSIS - - - - - - - - - - 11

B. OBJECTIVES OF THIS STUDY - - - - - - - - - - - - - - - - - - 11

C. PURPOSE OF THE PRESENTATION - - - - - - - - - - - - - - - - 12

II. THE FINITE ELEMENT FORMULATION - - - - - - - - - - - - - - - - 13

A. FORMULATION OF THE VARIATIONAL METHOD - - - - - - - - - - - 13

B. FORMULATION OF THE FINITE ELEMENT PROBLEM - - - - - - - - - 15

C. BOUNDARY CONDITIONS - 17

D. THE MATRIX EQUATION - 18

E. FORMULATION OF THE ELEMENT MATRIX - - - - - - - - - - - - - 20

F. LUMPED ELEMENT PROPERTIES - - - - - - - - - - - - - - - - - 24

III. PROGRAM STRUCTURE - 27

A. PROGRAM IDENTIFICATION - - - - - - - - - - - - - - - - - - - 27

B. PURPOSE - 27

C. PROGRAMING INFORMATION - - - - - - - - - - - - - - - - - - - 27

D. PROGRAM CAPACITY - 27

E. GENERAL PROGRAM STRUCTURE - - - - - - - - - - - - - - - - - 28

F. MAIN PROGRAM - 34

1. Control Information - - - - - - - - - - - - - - - - - - - 34

2. Material Properties - - - - - - - - - - - - - - - - - - - 36

3. Time Sequencing Information - - - - - - - - - - - - - - - 36

4. Nodal Point Information - 36

5. Element Information - 38

6. Time Steps - 38

7. Convection Boundary Conditions - - - - - - - - - - - - - - - 39

8. Temperature and Flux Boundary Conditions - - - - - - - - 40

9. Temperature Output - 40

G. SUBROUTINE FORM - 40

H. SUBROUTINE SYMSOL - 43

I. TEMPERATURE DEPENDENT PROPERTIES - - - - - - - - - - - - - - - 45

IV. USER INFORMATION - 46

A. INPUT DATA - 46

B. CONTROL PARAMETERS - 46

C. MATERIAL PROPERTIES - 47

D. TIME SEQUENCING - 47

E. NODAL POINT CONSTRUCTION - - - - - - - - - - - - - - - - - - - 48

F. ELEMENT CONSTRUCTION - 51

G. BOUNDARY CONDITIONS - 52

H. UNITS - 53

I. ERROR EXITS - 54

J. COORDINATES - 54

K. TEMPERATURE OUTPUT - 54

L. INPUT DATA PREPARATION - 54

1. Control Information - 55

2. Material Properties - 55

3. Time Sequencing Information - - - - - - - - - - - - - - - - 56

4. Nodal Point Information - - - - - - - - - - - - - - - - - - - 56

5. Element Information - 56

6. Convection Boundary Conditions - - - - - - - - - - - - - - 57

7. Temperature and Flux Boundary Conditions - - - - - - - - 57

8. Multiple Problems - 57

RECOMMENDATIONS - 58

APPENDIX A EXAMPLE PROBLEMS - 59

APPENDIX B COMPUTER PROGRAM VARIABLES - - - - - - - - - - - - - - - - - - - 64

APPENDIX C SAMPLE DATA CARDS FOR EXAMPLE 3 - - - - - - - - - - - - - - - 66

COMPUTER PROGRAM - 68

COMPUTER OUTPUT - 81

BIBLIOGRAPHY - 84

INITIAL DISTRIBUTION LIST - 85

FORM DD 1473 - 87

LIST OF FIGURES

2-1 Triangular Element - 20

2-2 Quadrilateral Elemement - 24

3-1 Main Program Flow Diagram, Part I - - - - - - - - - - - - - - 29

3-2 Main Program Flow Diagram, Part II - - - - - - - - - - - - - - 30
Constant Properties and Boundary Conditions

3-3 Main Program Flow Diagram, Part II - - - - - - - - - - - - - - 31
Temperature Dependent Properties and
Constant Boundary Conditions

3-4 Main Program Flow Diagram, Part II - - - - - - - - - - - - - - 32
Constant Properties and
Time Dependent Boundary Conditions

3-5 Main Program Flow Diagram, Part II - - - - - - - - - - - - - - 33
Temperature Dependent Properties and
Time Dependent Boundary Conditions

3-6 Main Program Flow Diagram, Part III - - - - - - - - - - - - - 35

3-7 Nodal Point and Element Generation - - - - - - - - - - - - - 37

3-8 The Matrix Equation - 41

3-9 Computer Storage for the Matrix of Coefficients - - - - - 42

3-10 The Matrix Equation in Upper Triangular Form - - - - - - - 44

4-1 Irregularly Bounded Configuration - - - - - - - - - - - - - - 49

4-2 Reduced Irregularly Bounded Configuration - - - - - - - - - 50

4-3 Conversion of a Right-Triangle to a Quadrilateral - - - - 51

A-1 One Dimensional Heat Conduction - - - - - - - - - - - - - - - 52

A-2 Two Dimensional Heat Conduction - - - - - - - - - - - - - - - 61

LIST OF SYMBOLS

c	Specific Heat
h	Convection Heat Transfer Coefficient
k	Thermal Conductivity
k_{ij}	Thermal Conductivity Tensor
$\dot{q}$	Heat Generation
q_i	Heat Flux Vector
n_i	Unit Normal Vector
S	Surface of a Region
T	Temperature
$\dot{T}$	First Time Derivative or Temperature
V	Volume of a Region
t	Time
Δt	Time Increment
α	Element Spacial Coefficients
α_T	Thermal Diffusivity
ϵ	A Small Parameter
λ	Arbitrary Family of Curves
Π	Functional of T and $\dot{T}$
ρ	Mass Density
Δ	Element Area
$[\;]$	Rectangular Matrix
$\{\;\}$	Column Vector
$\langle\;\rangle$	Row Vector
$[\;]^T$	Matrix Transpose
$[\;]^{-1}$	Matrix Inversion

ACKNOWLEDGEMENTS

The computer program described in this paper was modified during the author's graduate study for the Master of Science degree in Mechanical Engineering at the United States Naval Postgraduate School.

The author wishes to extend his thanks to Dr. Robert E. Nickell, of Bell Aero Systems, who was the original author of the program. Also many thanks are extended to Professor Gilles Cantin of the Naval Postgraduate School for his assistance on this work. The author would finally like to express his gratitude to Professor Paul Pucci of the Naval Postgraduate School for his constant encouragement and helpfulness during the supervision of this work.

The Naval Postgraduate School Computer Center provided the facilities for the computer work.

I. INTRODUCTION

In this chapter the reasons and advantages of the finite element analysis method are given. The purpose of this presentation and the objectives of the study are discussed.

A. REASONS FOR THE FINITE ELEMENT ANALYSIS

With the ever increasing complexity of problems occuring from engineering situations in design and analysis, there is a rising need for an approximate method of solution for these problems. This technique must be flexible enough to encompass a large variety of problems and still give accurate answers to the simplest of situations. Since most approximate methods of analysis of complex engineering problems involve many tedious calculations, it would enhance the usefulness of the technique if it were capable of being programed for computer solution. This would relieve the engineer of the tedious burden of hand calculated solutions and leave him more time to devote to the job of engineering.

The finite element method of analysis is an approximate method of analyzing engineering problems which meet these specifications. It has a high degree of flexibility and can be readily applied to computer solutions to obtain rapid solutions to complex problems.

B. OBJECTIVES OF THIS STUDY

The major objective of this study is a computer program using the finite element method of analysis, for the solution of two-dimensional unsteady heat conduction problems. In fulfilling this goal there were several minor objectives accomplished along the way. The original program was written for the IBM 7094 system. It was converted for operation on the IBM/OS 360 computer at the Naval Postgraduate School Computer Center.

Involved in this conversion was the changing of control cards and a few FORTRAN IV language differences. Also faster random access disks were substituted for the magnetic tapes originally used for temporary storage. The input and output formats were overhauled to make better use of space on the printed output page. More extensive labels were added to the printed output in order to ease the use of the program by persons not familiar with the finite element technique.

C. PURPOSE OF THE PRESENTATION

The purpose of this presentation is to make available a computer program using the finite element method of analysis for the solution of unsteady heat conduction problems. In doing so an introduction into the formulation of the finite element method is given for those not familiar with the technique, to give an indication of the power of this type of analysis. A description of the components of the program is included to enable the user to better understand the logic of the computer solution and increase the user's ability to formulate problems to which the program can be applied. Specific instructions on formulation of problems and loading of input data is made available in a chapter on user information. Finally, as is customary in works of this type, recommendations for future modifications of the program and possible applications are included.

II. THE FINITE ELEMENT FORMULATION

In this chapter the finite element method is formulated. The idea of a functional of a function and the variation of a functional are introduced. These principles are developed into an approximate solution of the two-dimensional unsteady heat conduction equation using a matrix iterative solution.

A. FORMULATION OF THE VARIATIONAL METHOD

If a functional of a temperature field can be found such that its extremum yields an equation describing unsteady heat conduction,

$$k_{ij} T_{,ji} = \rho c \dot{T} - \dot{q} \tag{2-1}$$

and an equation describing the heat flux across the surface of the region,

$$n_i k_{ij} T_{,j} = n_i q_i \tag{2-2}$$

then the variation of the functional will yield a set of first order, ordinary differential equations in terms of the nodal temperatures. Using the method of tensor notation, a comma denotes differentiation with respect to the following subscript and repeated subscripts imply summation.

Let Π be a functional of the temperature field T(x,y,z,t) and the first time derivative of the temperature field $\dot{T}$(x,y,z,t), be defined by

$$\Pi(T,\dot{T}) = \int_V \left(\tfrac{1}{2} T_{,i} k_{ij} + \rho c T \dot{T} - \dot{q} T\right) dV + \int_S n_i q_i T \, dS \tag{2-3}$$

The variation of $\Pi(T,\dot{T})$ with respect to T (with $\dot{T}$ held constant) is given by

$$\delta\Pi = \frac{\delta\Pi(T+\epsilon\lambda, \dot{T})}{\delta\epsilon} \tag{2-4}$$

where ϵ is a small parameter and λ is one of a family of functions. λ is zero on the boundary of the region and arbitrary elsewhere. An extremum of the functional $\Pi(T,\dot{T})$ implies that $\delta\Pi(T,\dot{T})$ equals zero, i.e.,

$$\left.\frac{\delta\Pi(T+\epsilon\lambda,\dot{T})}{\delta\epsilon}\right|_{\epsilon=0} = 0 \qquad (2\text{-}5)$$

First

$$\Pi(T+\epsilon\lambda,\dot{T}) = \int_V \Big(\tfrac{1}{2}(T+\epsilon\lambda)_{,i}\,k_{ij}(T+\epsilon\lambda)_{,j}$$

$$+\rho c(T+\epsilon\lambda)\dot{T} - \dot{q}(T+\epsilon\lambda)\Big)dv - \int_S n_i q_i (T+\epsilon\lambda)\,dS \qquad (2\text{-}6)$$

Using the identity

$$\frac{\delta}{\delta\epsilon}\left[\frac{1}{2}(T+\epsilon\lambda)_{,i}\,k_{ij}(T+\epsilon\lambda)_{,j}\right] = \left[(T+\epsilon\lambda)_{,i}\,k_{ij}\right]_{,j} - k_{ij}(T+\epsilon\lambda)_{,ji}\,\lambda$$

if the conductivity tensor is symmetric i.e. $k_{ij} = k_{ji}$. Then

$$\frac{\delta\Pi(T+\epsilon\lambda,\dot{T})}{\delta\epsilon} = \int_V \Big(\left[(T+\epsilon\lambda)_{,i}\,k_{ij}\,\lambda\right]_{,j} - k_{ij}(T+\epsilon\lambda)_{,ji}\,\lambda$$

$$+\rho c\lambda\dot{T} - \dot{q}\lambda\Big)dv - \int_S n_i q_i\,\lambda\,dS \qquad (2\text{-}7)$$

The volume integral $\int_V (T_{,i}\,k_{ij}\lambda)_{,j}\,dv$ can be transformed into a surface integral.

$$\int_V (T_{,i}\,k_{ij}\,\lambda)_{,j}\,dv = \int_S n_i k_{ij} T_{,j}\,\lambda\,dS \qquad (2\text{-}8)$$

When (2-7) is evaluated at $\epsilon = 0$

$$\delta\pi(T+\epsilon\lambda,\dot{T}) = \int_V \Big((T_{,i} k_{ij} \lambda)_{,j} - k_{ij} T_{,ji} \lambda + \rho c \lambda \dot{T} - \dot{q} \lambda \Big) dv - \int_S n_i q_i \lambda ds$$

or

$$\delta\pi(T+\epsilon\lambda,\dot{T}) = \int_V (-k_{ij} T_{,ji} + \rho c \dot{T} - \dot{q}) \lambda dv + \int_S n_i k_{ij} T_{,j} \lambda dS - \int_S n_i q_i \lambda ds \tag{2-9}$$

The extremum of π $(\delta\pi = 0)$ reduces to

$$k_{ij} T_{,ji} = \rho c \dot{T} - \dot{q} \tag{2-10}$$

in the region V. Also on the surface S

$$n_i k_{ij} T_{,j} = n_i q_i \tag{2-11}$$

These two equations (2-10, 2-11) are the unsteady heat conduction equation and the boundary flux equation. This verifies that a function T which yields an extremum of the functional defined by (2-3) satisfies both the unsteady heat conduction equation in the region V and the boundary flux equation on the surface S [6]*.

B. FORMULATION OF THE FINITE ELEMENT PROBLEM

If the choice of T and $\dot{T}$ is restricted so that they are represented by certain constants which are obtained in their formulation, the functional π becomes a real-valued function. In this case π shall be $\pi(\{T\},\{\dot{T}\})$ where $\{T\}$ and $\{\dot{T}\}$ are vectors of nodal point values of $\{T\}$ and $\{\dot{T}\}$. The extremum of $\pi(\{T\},\{\dot{T}\})$ must now be found which

---

*Numbers in brackets refer to similarly numbered items in Bibliography.

implies

$$\frac{\delta\pi(\{T\},\{\dot{T}\})}{\delta T_i} = 0 \tag{2-12}$$

From this point on the finite element formulation will be considered in two dimensions only. The region V will be divided into a number plane triangular elements. The functions T and $\dot{T}$ can be uniquely described within each of these elements by the values of the function at each of the nodal points of the element T_i, T_j, T_m, and linear function of the coordinates of the element.

Let the temperature field in an element be given by

$$T(x,y,t) = \langle\phi(x,y)\rangle[A]\{T_\xi(t)\} \quad \xi = i,j,m \tag{2-13}$$

and the time rate of temperature change given by

$$\dot{T}(x,y,t) = \langle\phi(x,y)\rangle[A]\{\dot{T}_\xi(t)\} \tag{2-14}$$

$\langle\phi\rangle$ is a vector which specifies the spacial approximation of T and $\dot{T}$. The matrix $[A]$ is a constant and is defined by the above relationships. Its exact value and derivation will be discussed in detail in section II, E.

The temperature gradient $T_{,\eta}$ (η = x,y) can be expressed in terms of the nodal temperatures by

$$\begin{Bmatrix} T_{,x} \\ T_{,y} \end{Bmatrix} = [D]\langle\phi\rangle[A]\{T_\xi\} = [\phi'][A]\{T_\xi\} \tag{2-15}$$

In the following development the subscript ξ, denoting nodal point values, will be dropped and will be assumed.

Writing the functional Π in terms of nodal point values,

$$\Pi(\{T\},\{\dot{T}\}) = \int_v \Big(\tfrac{1}{2}\{T\}^T[A]^T[\phi']^T[k][\phi'][A]\{T\} + \rho c\{T\}^T[A]^T\langle\phi\rangle^T\langle\phi\rangle[A]\{\dot{T}\}$$

$$- \{T\}^T[A]^T\langle\phi\rangle^T\dot{q}\Big)dv - \int \{T\}^T[A]^T\langle\phi\rangle^T\{n\}\{q\}dS \qquad (2\text{-}16)$$

Taking the variation of Π with respect to $\{T\}$ and setting it equal to zero

$$\delta\pi(\{T\},\{\dot{T}\}) = [K]\{T\} - \{\dot{q}^*\} + [C]\{\dot{T}\} - \{q^*\} = 0 \qquad (2\text{-}17)$$

where

$$[K] = \int_v [A]^T[\phi']^T[k][\phi'][A]dv \qquad (2\text{-}18)$$

$$[C] = \int_v \rho c[A]^T\langle\phi\rangle^T\langle\phi\rangle[A]dv \qquad (2\text{-}19)$$

$$[\dot{q}^*] = \int_v \dot{q}[A]^T\langle\phi\rangle^T dv \qquad (2\text{-}20)$$

$$\{q^*\} = \int_S [A]^T\langle\phi\rangle^T\{n\}\{q\}dS \qquad (2\text{-}21)$$

C. BOUNDARY CONDITIONS

There are four types of boundary conditions to be considered. (1) specified temperature, (2) specified flux, (3) convection boundary layer, and (4) adiabatic. For a specified temperature boundary condition T_i is constant over boundary segment S_1. For a specified flux $q = \overline{q}$ over boundary segment S_2. For a convection boundary layer $q = h(T_i - T_o)$ over boundary segment S_3. Finally for an adiabatic boundary condition $q = 0$ over boundary segment S_4.

The boundary integral (2-21) becomes

$$\{q^*\} = \{\bar{q}^*\} + [H]\{T\} - \{h^*\} \tag{2-22}$$

where

$$\{\bar{q}^*\} = \int_{S_2} [A]^T \langle \phi \rangle^T \{n\}\{q\}\, ds \tag{2-23}$$

$$[H] = h \int_{S_3} [A]^T \langle \phi \rangle^T \langle \phi \rangle [A]\, ds \tag{2-24}$$

$$\{h^*\} = h T_0 \int_{S_3} [A]^T \langle \phi \rangle^T ds \tag{2-25}$$

The integral on segment S_1 is zero because the variation of the functional was specified as zero on the boundary. The integral on segment S_4 is zero since there is no heat flow across the boundary.

To obtain the extremum of the functional Π the nodal point temperatures $\{T\}$ which satisfy the following matrix equation must be found [6].

$$([k]-[H])\{T\} + [C]\{\dot{T}\} - \{\bar{q}^*\} - \{\dot{q}^*\} + \{h^*\} = 0 \tag{2-26}$$

D. THE MATRIX EQUATION

Let the time variable be t_i such that $t_i = i\Delta t$ $\quad i = 0,1,2,\ldots$

For this development i will refer to time increments.

Let

$$[K] = [k] - [H] \tag{2-27}$$

Equation (2-17) can be written as

$$[K]\{T\}_i + [C]\{\dot{T}\}_i = \{f\}_i \tag{2-28}$$

where $$\{f\}_i = \{\dot{g}^*\} + \{\bar{g}^*\} - \{h^*\} \tag{2-29}$$

Assuming that $\{\ddot{T}\}$ is constant over the interval $t_i \leq t \leq t_{i+1}$, i.e.,

$$\{\ddot{T}\}_i = \left(\{\dot{T}\}_{i+1} - \{\dot{T}\}_i\right)/\Delta t$$

$\{T\}_{i+1}$ can be obtained by a Taylor's expansion around $t = t_i$.

$$\{T\}_{i+1} = \{T\}_i + \Delta t\{\dot{T}\}_i + \frac{\Delta t^2}{2}\{\ddot{T}\}_i \tag{2-30}$$

$$\{T\}_{i+1} = \{T\}_i + \Delta t\{\dot{T}\}_i + \frac{\Delta t}{2}\left(\{\dot{T}\}_{i+1} - \{\dot{T}\}_i\right) \tag{2-31}$$

$$\{T\}_{i+1} = \{T\}_i + \frac{\Delta t}{2}\left(\{\dot{T}\}_{i+1} + \{\dot{T}\}_i\right) \tag{2-32}$$

Using equations (2-29) and (2-32) $\{T\}_{i+1}$ and $\{\dot{T}\}_{i+1}$ can be determined in terms of $\{T\}_i$ and $\{\dot{T}\}_i$. Solving (2-32) for $\{\dot{T}\}_{i+1}$ gives

$$\{\dot{T}\}_{i+1} = \frac{2}{\Delta t}\left(\{T\}_{i+1} - \{T\}_i\right) - \{\dot{T}\}_i \tag{2-33}$$

Substituting this into (2-28) gives

$$[K] + \frac{2}{\Delta t}[C]\{T\}_{i+1} = \{f\}_{i+1} + [C]\left(\frac{2}{\Delta t}\{T\}_i + \{\dot{T}\}_i\right) \tag{2-34}$$

Also from equation (2-28)

$$\{\dot{T}\}_i = [C]^{-1}\{f\} - [C]^{-1}[K]\{T\}_i \tag{2-35}$$

Substituting (2-34) into (2-35) gives

$$\left([K] + \frac{2}{\Delta t}[C]\right)\{T\}_{i+1} = \left(\frac{2}{\Delta t}[C] - [K]\right)\{T\}_i + \{f\}_i + \{f\}_{i+1} \tag{2-36}$$

or

$$\left([K] + \frac{2}{\Delta t}[C]\right)\left(\frac{\{T\}_{i+1} + \{T\}_i}{2}\right) = \frac{2}{\Delta t}[C]\{T\}_i + \left(\frac{\{f\}_i + \{f\}_{i+1}}{2}\right) \tag{2-37}$$

In a multi-element body the element matrix equations must be assembled into a single matrix equation. The assembly and solution of the equation is a complex task but it can be greatly simplified by the use of a computer [6].

E. FORMULATION OF THE ELEMENT MATRIX

If the region to be analyzed is divided into plane triangular elements then a typical element would appear like the following figure.

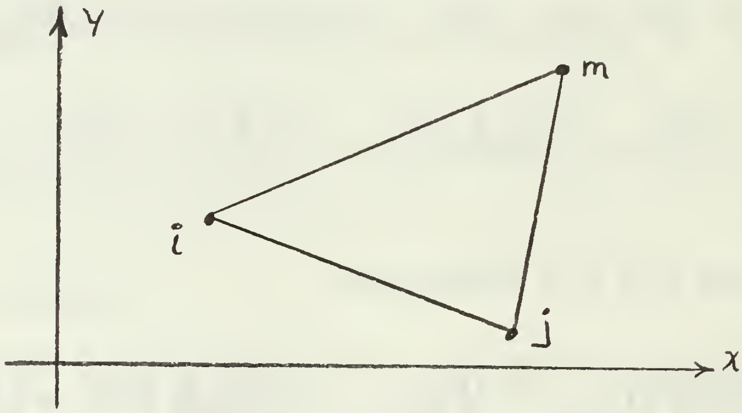

Figure 2-1 Triangular Element

The temperature in an element can be described in terms of a linear combination of the coordinates of the elements nodal points and the values of the temperature at each of the nodal points.

Assuming that the temperature distribution is of the form

$$T(x,y,t) = \alpha_1(t) + \alpha_2(t)x + \alpha_3(t)y \tag{2-38}$$

The temperature at each of the nodal points can be written as

$$
\begin{aligned}
T_i &= \alpha_1 + x_i\alpha_2 + y_i\alpha_3 \\
T_j &= \alpha_1 + x_j\alpha_2 + y_j\alpha_3 \qquad (2\text{-}39) \\
T_m &= \alpha_1 + x_m\alpha_2 + y_m\alpha_3
\end{aligned}
$$

Solving for the coefficients α_1, α_2, and α_3 using Cramer's rule the following results are obtained.

LET

$$\begin{vmatrix} 1 & x_i & y_i \\ 1 & x_j & y_j \\ 1 & x_m & y_m \end{vmatrix} = 2\Delta \qquad (2\text{-}40)$$

Note: is the area of the triangular element.

$$\alpha_1 = \begin{vmatrix} T_i & x_i & y_i \\ T_j & x_j & y_j \\ T_m & x_m & y_m \end{vmatrix} \Big/ 2\Delta$$

$$\alpha_2 = \begin{vmatrix} 1 & T_i & y_i \\ 1 & T_j & y_j \\ 1 & T_m & y_m \end{vmatrix} \Big/ 2\Delta$$

$$\alpha_3 = \begin{vmatrix} 1 & x_i & T_i \\ 1 & x_j & T_j \\ 1 & x_m & T_m \end{vmatrix} \Big/ 2\Delta$$

$$\alpha_1 = \left(T_i\begin{vmatrix} x_j & y_j \\ x_m & y_m \end{vmatrix} + T_j\begin{vmatrix} x_m & y_m \\ x_i & y_i \end{vmatrix} + T_m\begin{vmatrix} x_i & y_i \\ x_j & y_j \end{vmatrix}\right)/2\Delta$$

$$\alpha_2 = \left(T_i\begin{vmatrix} 1 & y_m \\ 1 & y_j \end{vmatrix} + T_j\begin{vmatrix} 1 & y_i \\ 1 & y_m \end{vmatrix} + T_m\begin{vmatrix} 1 & y_j \\ 1 & y_i \end{vmatrix}\right)/2\Delta$$

$$\alpha_3 = \left(T_i\begin{vmatrix} 1 & x_j \\ 1 & x_m \end{vmatrix} + T_j\begin{vmatrix} 1 & x_m \\ 1 & x_i \end{vmatrix} + T_m\begin{vmatrix} 1 & x_i \\ 1 & x_j \end{vmatrix}\right)/2\Delta$$

Recalling that the temperature distribution was previously described by the expression

$$T = \langle\phi\rangle[A]\{T\} \tag{2-42}$$

$\{T\}$ is the vector of the nodal temperatures and $[A]$ is the spacial dependence matrix which is a constant. Let

$$\langle\phi\rangle = \langle 1 \quad x \quad y\rangle \tag{2-43}$$

$$\{T\} = \begin{Bmatrix} T_i \\ T_j \\ T_m \end{Bmatrix} \qquad [A] = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \alpha_{13} \\ \alpha_{21} & \alpha_{22} & \alpha_{23} \\ \alpha_{31} & \alpha_{32} & \alpha_{33} \end{bmatrix}$$

then

$$T = \langle\phi\rangle[A]\{T\}$$

so

$$T = \langle 1 \quad x \quad y \rangle \begin{Bmatrix} (\alpha_{11} T_i + \alpha_{12} T_j + \alpha_{13} T_m) \\ (\alpha_{21} T_i + \alpha_{22} T_j + \alpha_{23} T_m) \\ (\alpha_{31} T_i + \alpha_{32} T_j + \alpha_{33} T_m) \end{Bmatrix} \quad (2\text{-}43a)$$

$$T = (\alpha_{11} T_i + \alpha_{12} T_j + \alpha_{13} T_m) + (\alpha_{21} T_i + \alpha_{22} T_j + \alpha_{23} T_m)\, x + (\alpha_{31} T_i + \alpha_{32} T_j + \alpha_{33} T_m)\, y$$

Therefore, comparing (2-43a) with (2-41)

$$\alpha_{11} = \begin{vmatrix} x_j & y_j \\ x_m & y_m \end{vmatrix} = x_j y_m - x_m y_j$$

$$\alpha_{12} = \begin{vmatrix} x_m & y_m \\ x_i & y_i \end{vmatrix} = x_m y_i - x_i y_m$$

$$\alpha_{13} = \begin{vmatrix} x_i & y_i \\ x_j & y_j \end{vmatrix} = x_i y_j - x_j y_i$$

$$\alpha_{21} = \begin{vmatrix} 1 & y_m \\ 1 & y_j \end{vmatrix} = y_j - y_m$$

$$\alpha_{22} = \begin{vmatrix} 1 & y_i \\ 1 & y_m \end{vmatrix} = y_m - y_i$$

$$\alpha_{23} = \begin{vmatrix} 1 & y_j \\ 1 & y_i \end{vmatrix} = y_i - y_j$$

$$\alpha_{31} = \begin{vmatrix} 1 & x_j \\ 1 & x_m \end{vmatrix} = x_m - x_j$$

$$a_{32} = \begin{vmatrix} 1 & x_m \\ 1 & x_i \end{vmatrix} = x_i - x_m$$

$$a_{33} = \begin{vmatrix} 1 & x_i \\ 1 & x_j \end{vmatrix} = x_j - x_i$$

The spacial dependence matrix is then defined by

$$[A] = \frac{1}{2\Delta} \begin{bmatrix} (x_j y_m - x_m y_j) & (x_m y_i - x_i y_m) & (x_i y_j - x_j y_i) \\ y_j - y_m & y_m - y_i & y_i - y_j \\ x_m - x_j & x_i - x_m & x_j - x_i \end{bmatrix} \quad (2\text{-}44)$$

[7].

F. THE METHOD OF LUMPING ELEMENT PROPERTIES

When working with a computer it is usually more convenient to work with a quadrilateral element. The quadrilateral is divided into four triangles with the center nodal point common to each of the triangles. (see figure 2-2).

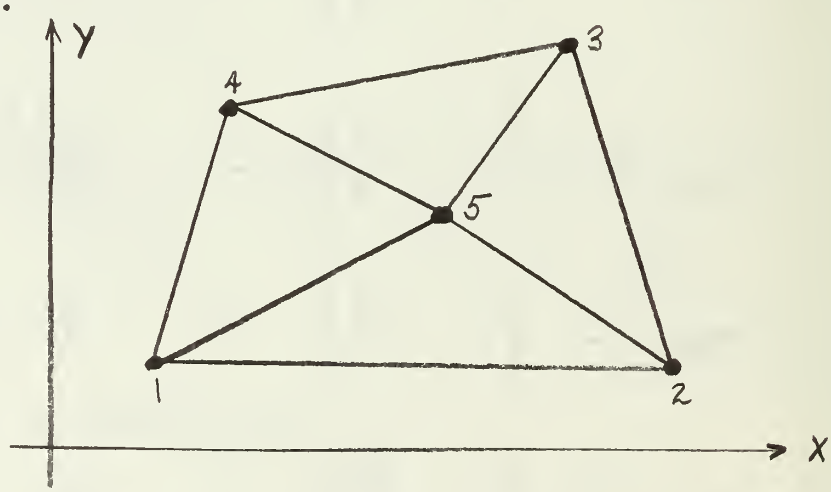

Figure 2-2 Quadrilateral Element

The matrix equation (2-28) can be assembled for this element in the following form

$$\begin{bmatrix} K_{11} & K_{12} & K_{13} & K_{14} & K_{15} \\ K_{21} & K_{22} & K_{23} & K_{24} & K_{25} \\ K_{31} & K_{32} & K_{33} & K_{34} & K_{35} \\ K_{41} & K_{42} & K_{43} & K_{44} & K_{45} \\ K_{51} & K_{52} & K_{53} & K_{54} & K_{55} \end{bmatrix} \begin{Bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{Bmatrix} + \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{15} \\ C_{21} & C_{22} & C_{23} & C_{24} & C_{25} \\ C_{31} & C_{32} & C_{33} & C_{34} & C_{35} \\ C_{41} & C_{42} & C_{43} & C_{44} & C_{45} \\ C_{51} & C_{52} & C_{53} & C_{54} & C_{55} \end{bmatrix} \begin{Bmatrix} \dot{T}_1 \\ \dot{T}_2 \\ \dot{T}_3 \\ \dot{T}_4 \\ \dot{T}_5 \end{Bmatrix} = \begin{Bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \end{Bmatrix}$$

Partitioning this equation results in

$$\begin{bmatrix} [K_{ij}] & \{K_{i5}\} \\ \langle K_{5j} \rangle & K_{55} \end{bmatrix} \begin{Bmatrix} \{T_i\} \\ T_5 \end{Bmatrix} + \begin{bmatrix} [C_{ij}] & \{C_{i5}\} \\ \langle C_{5j} \rangle & C_{55} \end{bmatrix} \begin{Bmatrix} \{\dot{T}_i\} \\ \dot{T}_5 \end{Bmatrix} = \begin{Bmatrix} \{f_i\} \\ f_5 \end{Bmatrix} \qquad (2\text{-}45)$$

$i,j = 1,2,3,4$

This equation can be multiplied out to yield two equations

$$[K_{ij}]\{T_i\} + \{K_{i5}\} T_5 + [C_{ij}]\{\dot{T}_i\} + \{C_{15}\} \dot{T}_5 = \{f_i\} \qquad (2\text{-}46)$$

$$\langle K_{5j} \rangle \{T_i\} + K_{55}T_5 + \langle C_{5j} \rangle \{\dot{T}_i\} + C_{55}\dot{T}_5 = f_5 \qquad (2\text{-}47)$$

The specific heat matrix $[C]$ is approximated by lumping the heat capacities at the four external nodes. Thus $[C]$ becomes a diagonal matrix (4x4) and $C_{55} = 0$. Therefore (2-46) can be written as

$$[K_{ij}]\{T_i\} + \{K_{i5}\} T_5 + [C_{ij}]\{\dot{T}_i\} = \{f_i\} \qquad (2\text{-}48)$$

and (2-47) as

$$\langle K_{5j} \rangle \{T_i\} + K_{55}T_5 = f_5 \qquad (2\text{-}49)$$

Solving (2-49) for T_5 and substituting into (2-48) results in

$$[K_{ij}]\{T_i\} + \{K_{i5}\} K_{55}^{-1}\left(f_5 - \langle K_{5j} \rangle \{T_i\}\right) + [C_{ij}]\{\dot{T}_i\} = \{f_i\} \qquad (2\text{-}50)$$

or

$$\left(\left[K_{ij}\right] - \left\{K_{i5}\right\} K_{55}^{-1} \left\langle K_{5j}\right\rangle \right) \left\{T_i\right\} + \left[C_{ij}\right]\left\{\dot{T}_i\right\} = \left\{f_i\right\} - \left\{K_{i5}\right\} K_{55}^{-1} f_5 \quad (2\text{-}51)$$

Equation (2-51) is analogous to equation (2-28) except that $[K]$ and $[C]$ are now (4x4) matrices and $\{f\}$ is a (4x4) vector [6].

III. PROGRAM STRUCTURE

This chapter describes the general composition of the program and gives a description of the function of each section of the program.

A. PROGRAM IDENTIFICATION

The program was written by Robert E. Nickell while in the Department of Civil Engineering, University of California at Berkeley. It was formerly identified as W2498,W035--2--57343, Nickell. It was written in July 1968 and was revised by Allan B. Chaloupka in May 1969. Formulation was based upon a variational principle derived by M. E. Gurtin of Brown University [4].

The program has been renamed DFETHC-Finite Element Transient Heat Conduction.

B. PURPOSE

The purpose of the program is to produce high speed solutions to transient heat conduction problems. These problems may involve plane or axisymmetric geometry, non-homogeneous anisotropic material configurations, temperature dependent material properties, and time dependent boundary conditions.

C. PROGRAMING INFORMATION

The program was originally written in FORTRAN IV for the IBM 7094 computer. It was converted for use on the IBM OS/360 Model 67 computer. The program was also rewritten in double precision (REAL *8).

D. PROGRAM CAPACITY

The size problem that can be analyzed by this program is subject to the following limitations"

Maximum number of nodal points 500

Maximum number of quadrilateral elements 490

Maximum number of materials 25

Maximum difference of nodal point numbers

for the same element 31

These are not stringent limitations and they may be adjusted somewhat by the user to fit individual problems. The limits are changed by changing the appropriate DIMENSION and COMMON statements and one card in the segment that calculates the half-band width. The only real limitation placed upon the size of the problem that can be analyzed is the amount of core storage available in the computer used.

E. GENERAL STRUCTURE

The program consists of a main program and several subprograms used for repetitive operations. The main program can be divided into three parts. The first part is concerned with reading into the program and printing out the data describing each problem. It is portrayed by the flow diagram in figure 3-1.

The second portion of the main program is concerned with the routing of operations according to certain logical variables. These variables describe four possible combinations. They are constant material properties and constant boundary conditions, temperature dependent material properties and constant boundary conditions, constant material properties and time dependent boundary conditions, and temperature dependent material properties and time dependent boundary conditions. Each of these possibilities is portrayed by a flow diagram in figures 3-2 through 3-5.

The third portion of the main program is concerned with calculating the effective forcing vector, solving the matrix equation and printing the temperatures for each time increment. The operation is then directed

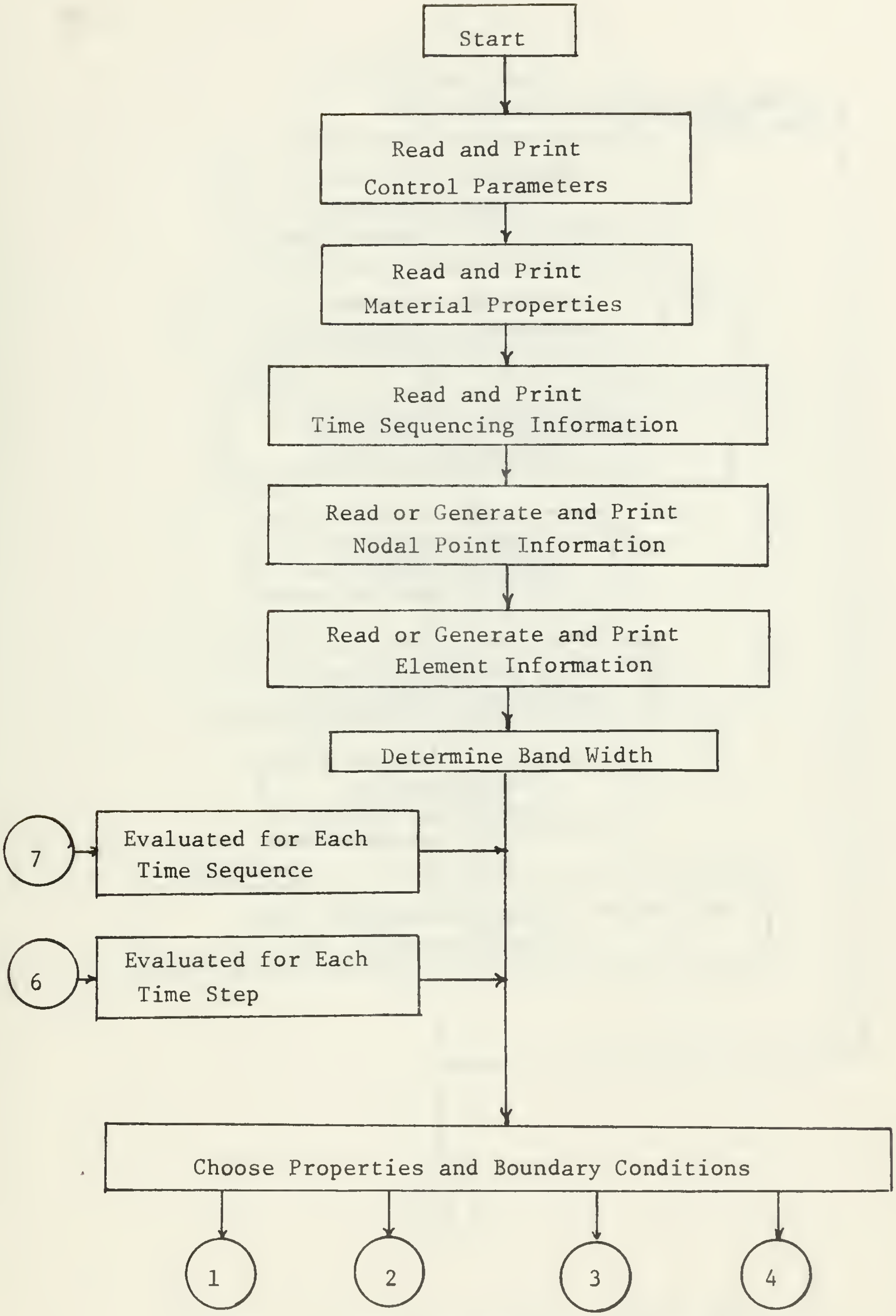

Figure 3-1 Main Program Flow Diagram, Part I

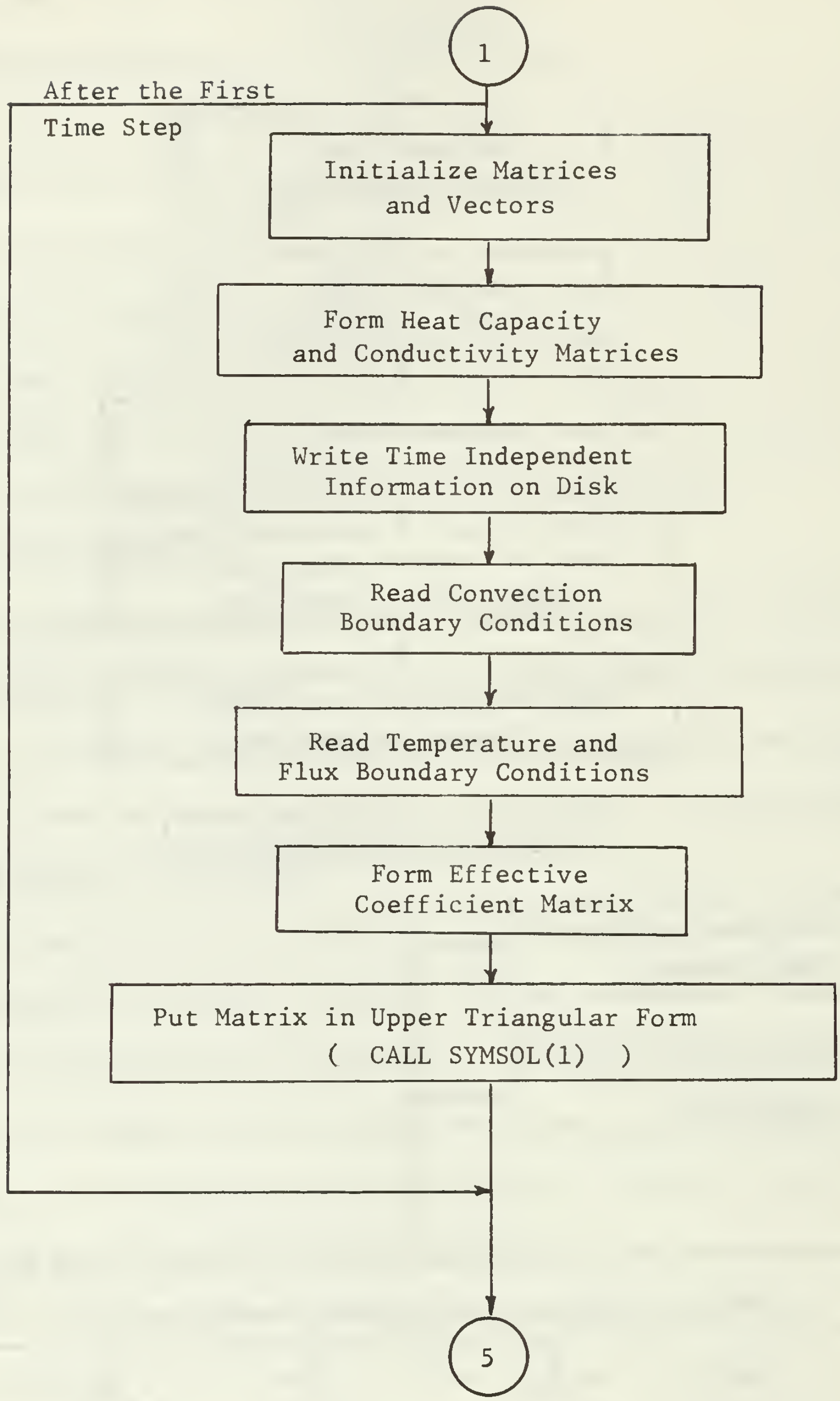

Figure 3-2 Main Program Flow Diagram, Part II
Constant Properties and Boundary Conditions

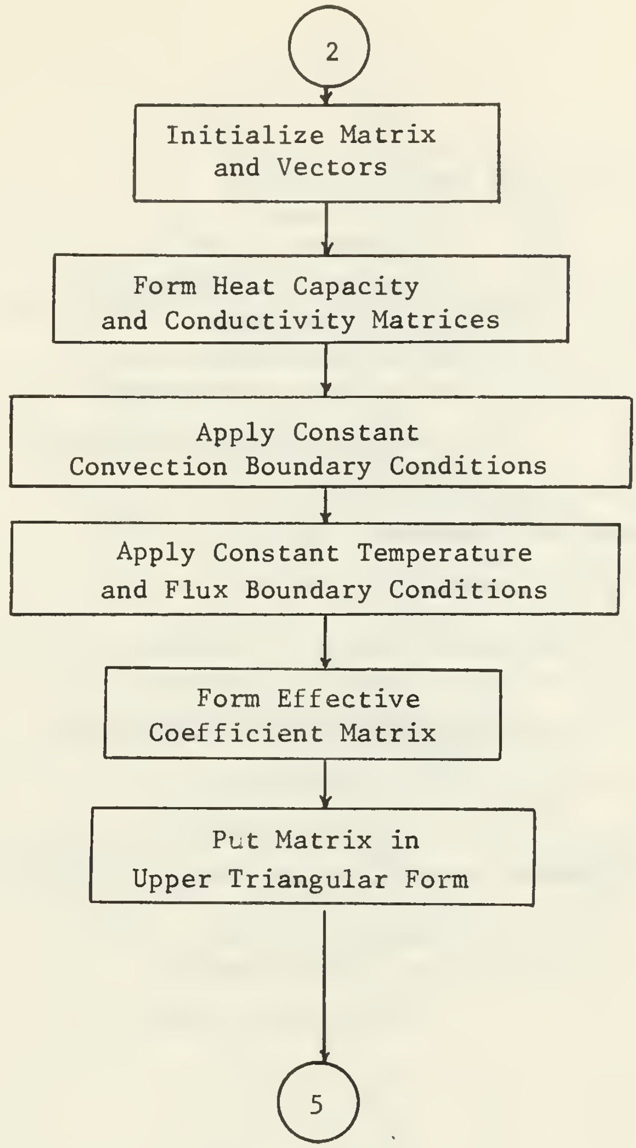

Figure 3-3 Main Program Flow Diagram, Part II
Temperature Dependent Properties and
Constant Boundary Conditions

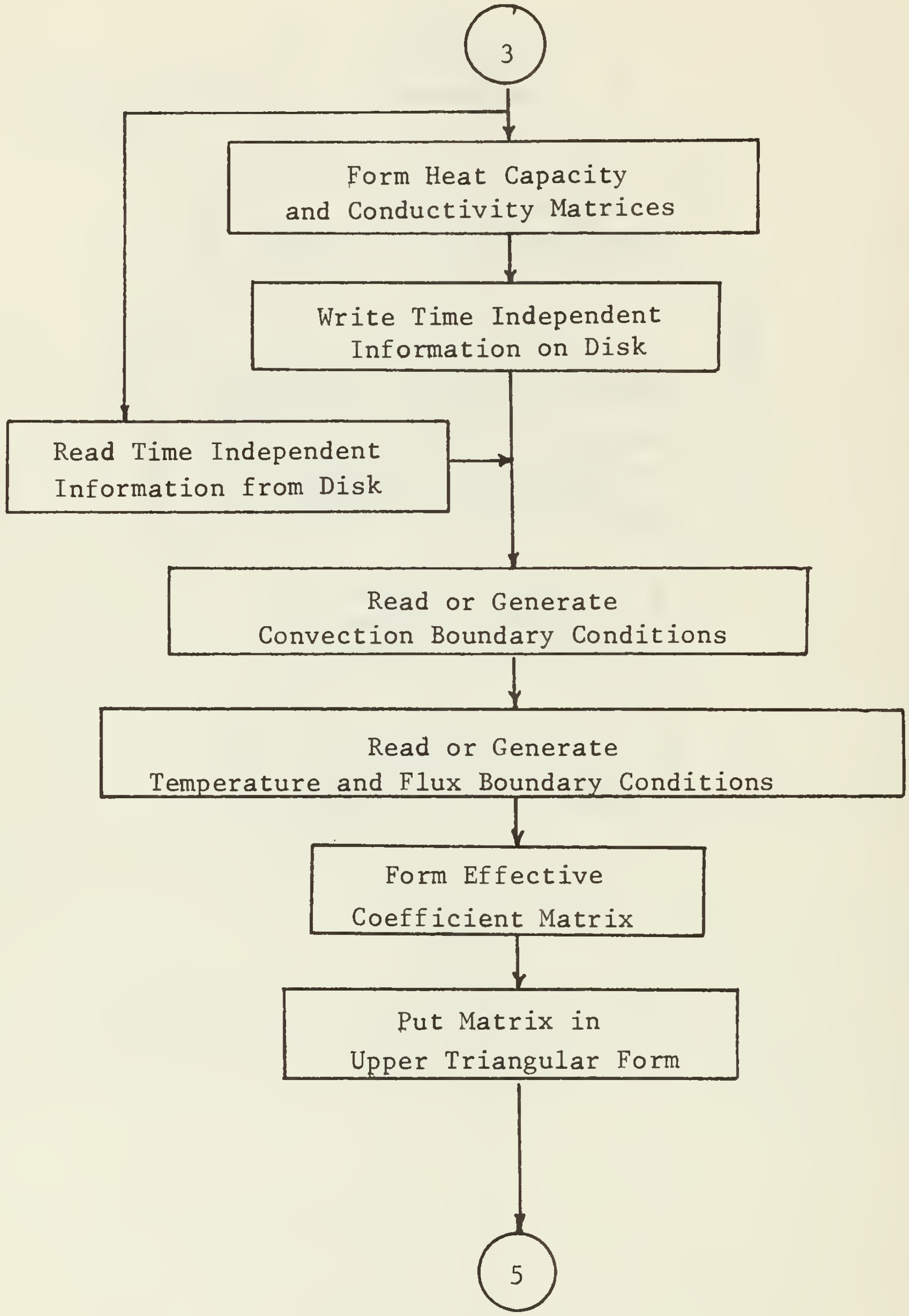

Figure 3-4 Main Program Flow Diagram, Part II
Constant Properties and Time Dependent Boundary Conditions

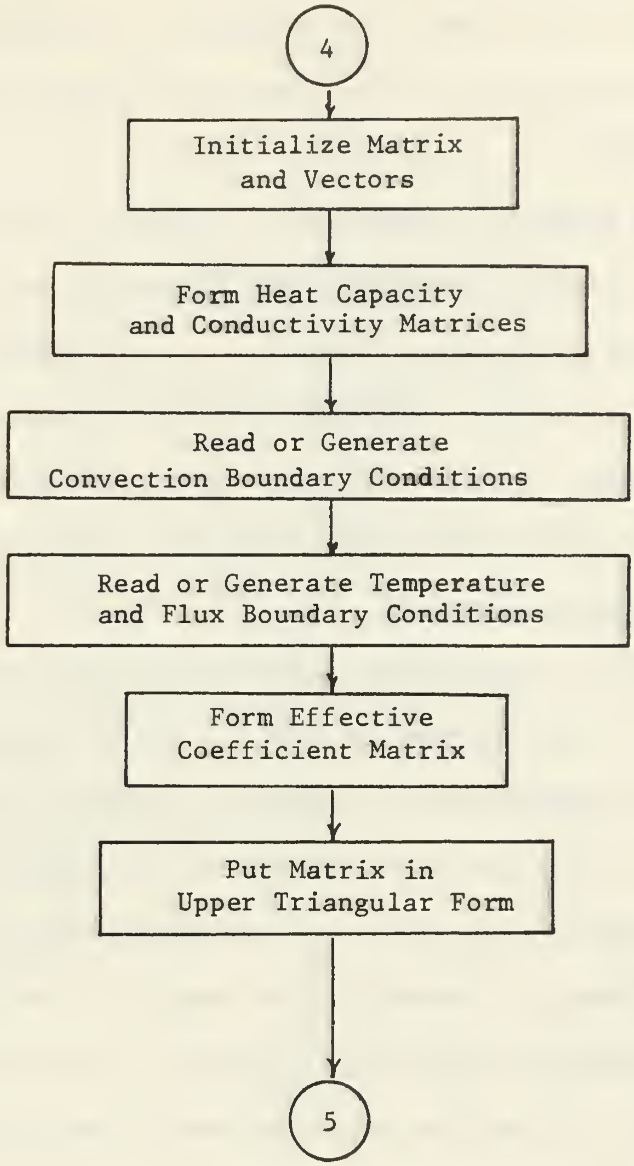

Figure 3-5 Main Program Flow Diagram, Part II
Temperature Dependent Properties and
Time Dependent Boundary Conditions

back to the beginning of the second section of the main program to start calculations for the next time increment. The third portion of the main program is portrayed by a flow diagram in figure 3-6.

F. MAIN PROGRAM

The main program is composed of a number of individual steps which comprise the finite element solution technique, and a system for the input and output of problem information. Not included in the main program are the matrix assembly processes, the matrix equation solving processes, and the boundary condition input routines. These are included in subprograms FORM, SYMSOL, TFUNC, CFUNC, FCBC, FTBC, and FFBC.

1. Control Information

The first information to be read into the program is the control information. This information sets the basic variables for each of the steps of the program and also routes the problem to various sections of the program. This information includes the number of nodal points, the number of elements, the number of convection boundary conditions, the number of materials, the number of time sequences, whether the problem is in plane or axisymmetric geometry, the output print interval, the system of units, whether a final spacial temperature distribution is desired, and the reference temperature of the body.

Also read in with the control information are two titles. The first is a 72 space title that can be used by the user to label each problem. The second is a 48 space title that is used to input the labels used with the system of units chosen.

The appropriate titles and control information are printed on the first page of the printed output, for each problem.

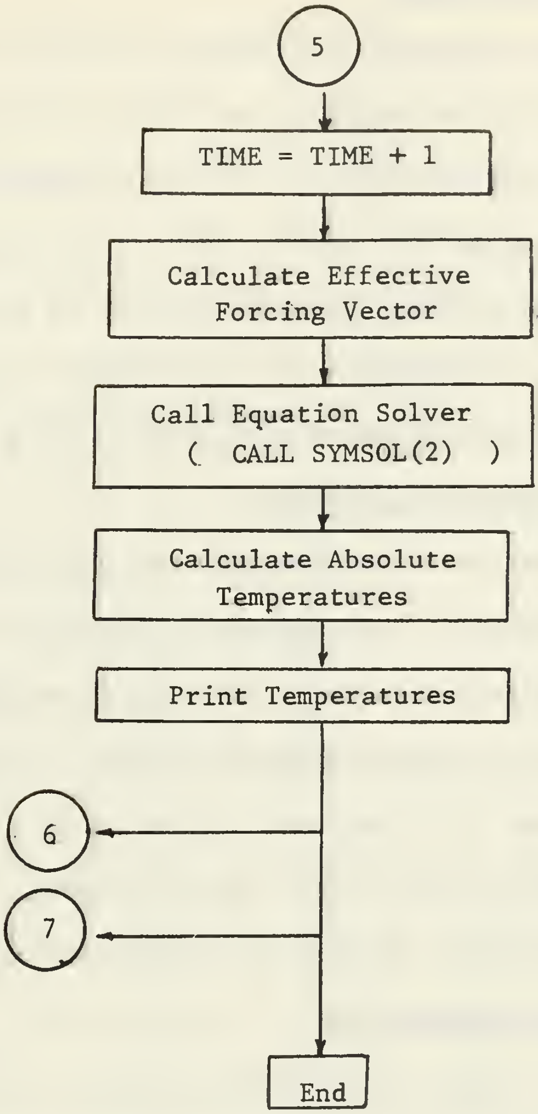

Figure 3-6 Main Program Flow Diagram, Part III

2. Material Properties

The thermal conductivity, density, specific heat, and heat generation properties for each material are read into the program. The material properties are then printed in order on the printed output page and labeled with the appropriate units. Since the program is for anisotropic materials, k_{xx}, k_{yy}, and k_{xy} are read in order to form the thermal conductivity tensor. The specific heat and density are read in as a product in order to reduce the number of variables in the program.

3. Time Sequencing Information

In this section the time sequencing information is read in for each sequencing scheme. This information consists of two logical variables and two numerical variables. The logical variables indicate whether there are temperature dependent properties and if there are time dependent boundary conditions. The two remaining variables describe the number of time steps and the size of each time increment. The time sequencing information is printed on the printed output page with appropriate labels.

4. Nodal Point Information

Ordinarily this segment of the program will read the x,y coordinates and initial temperature of each nodal point and print the information with the appropriate dimensions. Because of the possibility of a large number of nodal points (500) and the equally large possibility of human error in punching data cards there is a segment of the program that will generate nodal point information. This segment has a few limitations, which will be discussed in chapter IV, but it is usually able to generate almost all of the interior nodal points. As an example in the configuration shown in figure 3-7 the nodal points indicated by squares were read into the program and the remaining nodal points were generated by the program.

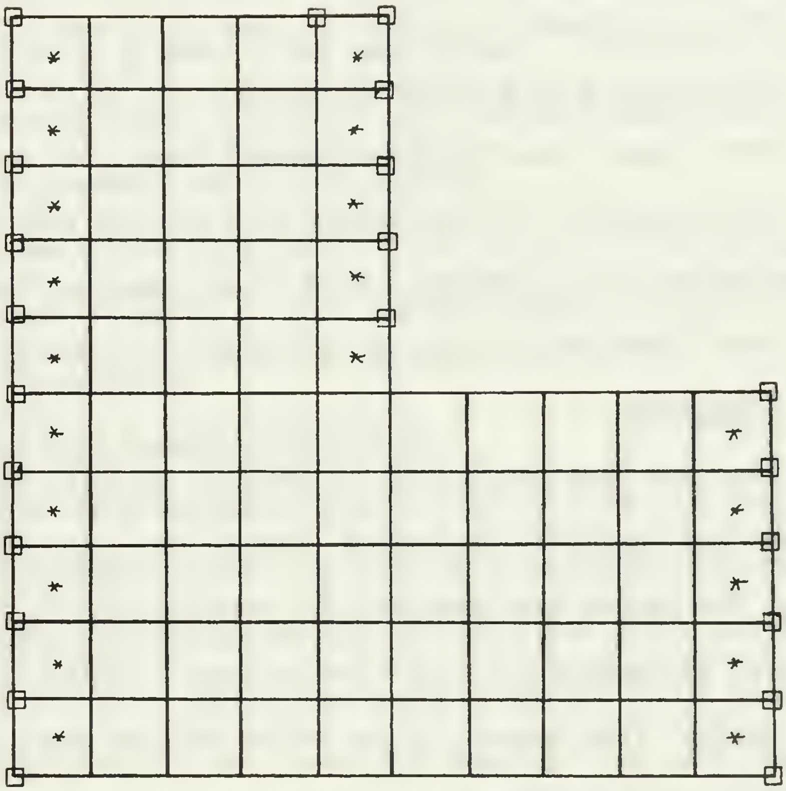

Figure 3-7 Nodal Point and Element Generation

5. Element Information

The purpose of this segment is about the same as the previous one. The number of the element, the number of each of its nodes, the type of material, and the amount of heat generation within the element are read into the program and printed out with appropriate labels. Again this segment has the capability to facilitate the loading of data by generating element information. This segment also has some limitations which will be discussed in chapter IV, but it is usually able to generate most of the elements in a given configuration. As an example, in figure 3-7 all the elements were generated except those that are starred.

Also included in this segment is a routine that calculates the half-band width of the problem. This is an important control parameter that is used later in the program to assemble the matrix equation.

6. Time Steps

The time step portion of the program includes two large loops. The first loop executes the program through each time sequence of the problem. The second loop executes the program through the required number of time increments for each time sequence. Inside these two loops the two logical time sequencing variables and the loop counters control the routing of the problem.

The two logical variables control the choice of variable material properties and boundary conditions. If the material properties and boundary conditions are constant then the coefficient matrix and forcing vector are calculated only once. The matrix equation is then solved for successive time increments using only the original matrices. If variable material properties are involved the effective coefficient matrix must be reapplied since the coefficient matrix is reformed from zero values. If time dependent boundary conditions are involved then the contribution of the boundary conditions to the coefficient matrix must be reapplied. The coefficient matrix from the original material properties is kept in temporary storage. When both time dependent boundary conditions and temperature dependent properties are used then the coefficient matrix and forcing vectors must be reformed for each time increment based upon the immediate time and temperature values and initial properties. The data for temperature dependent properties is calculated from the initial conditions and the present nodal temperatures. The data for time dependent boundary conditions is either read into the program with each time increment or generated within the program.

Different time sequences can be run for the same problem. This enables the user to speed or slow the time change during a particular segment of the problem.

7. Convection Boundary Conditions

A convection boundary condition occurs when the problem includes a convection boundary layer as part of its boundary conditions. One convection boundary condition consists of two nodal point numbers, the convective heat transfer coefficient and the ambient fluid temperature. The two nodal points define the convection boundary for each condition.

Time varying convection boundary conditions can be obtained by varying the ambient fluid temperature. The new fluid temperature may be read with each new time step or it may be generated by using the function subprogram FCBC. FCBC is used in the same manner as the function subprograms for temperature dependent material properties. The argument of FCBC is TIME and the value of FCBC is currently set at one. If temperature dependent material properties are used then the convection boundary condition section stores and reapplies the convection boundary condition contributions to the coefficient matrix and the forcing vector.

8. <u>Temperature and Flux Boundary Conditions</u>

A temperature or flux boundary condition occurs when the problem specifies a temperature or flux at a nodal point. One temperature or flux boundary conditions consists of the nodal point number and the value of the temperature or flux at that nodal point.

Time varying temperature and flux boundary conditions can be obtained by varying the temperature or flux at each nodal point. The new temperature or flux may be read with each new time step or it may be generated by using the function subprograms FTBC or FFBC for time dependent temperature or flux respectively. FTBC and FFBC are used in the same manner as the function subprograms for temperature dependent material properties. The argument of each is TIME. Their current values are both one. If temperature dependent properties are used the temperature and flux boundary condition section stores and reapplies the boundary conditions to the coefficient matrix and the forcing vector.

9. <u>Temperature Output</u>

This segment of the program is concerned with the print out of the temperature solutions for each nodal point at each time step. The value of the time is printed and the nodal point numbers and corresponding temperatures for that time are printed in rows below. If it is desired, the temperature can be printed with spacial data at the end of each time sequence.

G. SUBROUTINE FORM

The subroutine FORM is used to calculate the effective element thermal conductivity, heat capacity, and heat generation. The subroutine performs these operations for each of the elements of the configuration and, if temperature dependent properties are used, for each time increment.

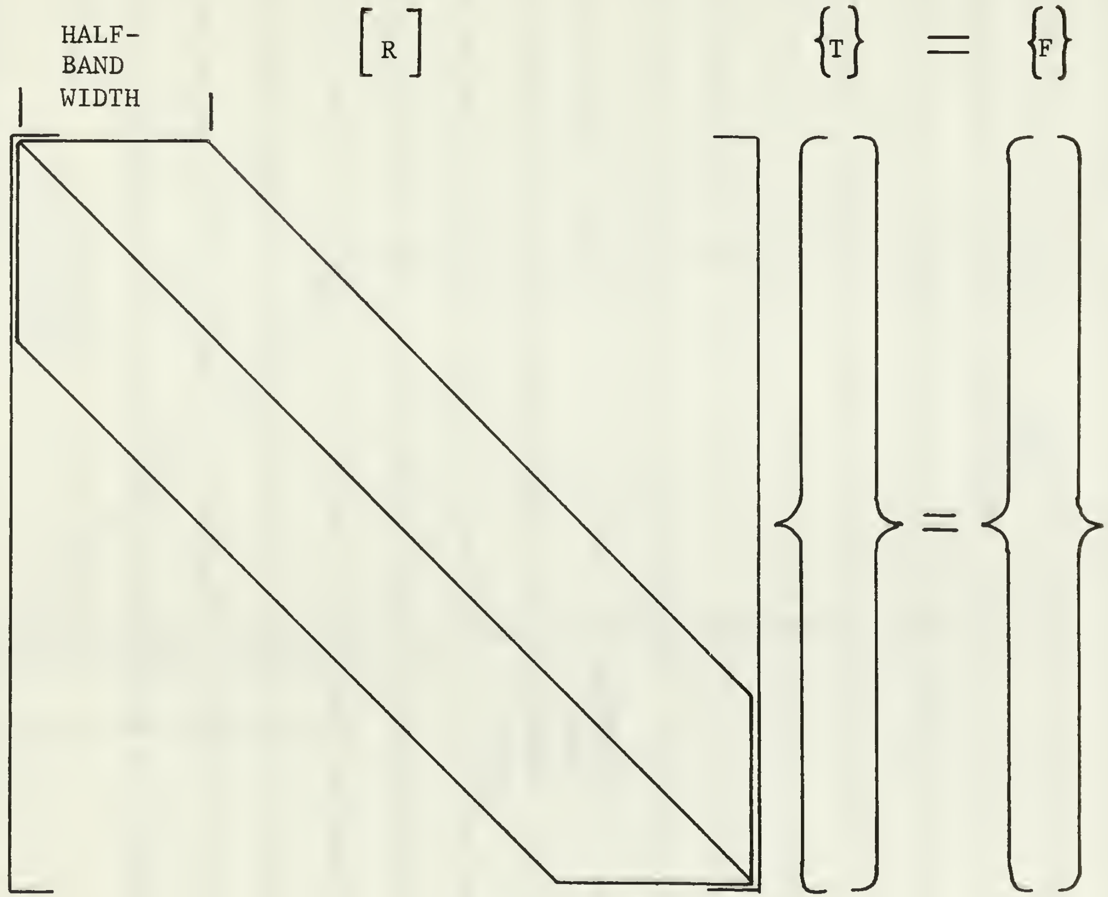

Figure 3-8 The Matrix Equation

Figure 3-9 Computer Storage for the
Matrix of Coefficients

First the mean x,y coordinates and the mean temperature of the element are calculated by averaging the values of each of the nodes. Using the mean x,y coordinates the quadrilateral element is divided into four triangles. The conductivity tensor, heat capacity, and heat generation vectors are formed for each of the four triangular sections. As these matrices are formed they are assembled into the larger (5) nodal point equation for the quadrilateral element. The heat capacity and heat generation vectors are assembled so that the fifth term of each vector is dropped when forming the lumped properties matrix equation. The center nodal point is eliminated from the thermal conductivity tensor (5x5), according to equation (2-52), to give the thermal conductivity tensor (4x4) for the lumped properties matrix equation. The thermal conductivity tensor and heat capacity and heat generation vectors for each element are returned to the main program where they are loaded into the coefficient matrix and vectors used in the solution of the total matrix equation for each time step. For problems where the properties of the materials are not temperature dependent the thermal conductivity tensor and the heat capacity and heat generation vectors are calculated only at the beginning of the problem. If temperature dependent material properties are included these matrices are calculated at the beginning of each time step.

H. SUBROUTINE SYMSOL

The subroutine SYMSOL in conjunction with the loading methods of the main program is used to solve the finite element matrix equation. It consists of two parts. The first part reduces the coefficient matrix to its final form. The second part forms the effective forcing vector and solves for the temperatures at each of the nodal points.

In an ordinary formulation the final matrix equation for the finite element method would look like figure 3-8. The matrix of coefficients [R] is a banded symmetric matrix. In the computer, in order to conserve core storage, the matrix of coefficients is stored in columns. The first column is the diagonal of the matrix of coefficients. (See figure 3-9). All the operations performed on the matrix of coefficients are done on the upper half of the symmetric banded matrix in its stored form.

After the coefficient matrix is formed it is put into upper triangular form. This yields a matrix equation of the form shown below.

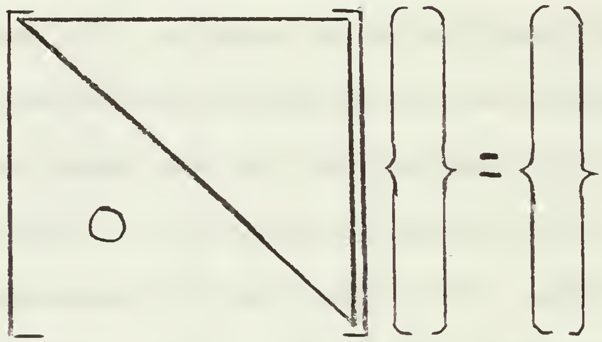

Figure 3-10 Matrix Equation in Upper Triangular Form

This is accomplished by the first segment of SYMSOL. The second portion of SYMSOL, using the effective forcing vector, starts with the temperature at the bottom of the nodal temperature vector and back substitutes to solve for the nodal temperatures. These operations are performed for each time increment.

I. TEMPERATURE DEPENDENT PROPERTIES

There are two function subprograms that are used in conjunction with temperature dependent properties. They are TFUNC and CFUNC. TFUNC is used for the temperature dependent thermal conductivity and CFUNC is used for the temperature dependent product of the density and specific heat. In the programs present configuration each of these function subprograms is prepared for temperature independent properties and their values are therefore equal to one. If the thermal conductivity were temperature dependent in the form of: $k = k_o(1 + bT)$, where b is a constant coefficient and k_o is the value of the thermal conductivity at $T = 0$, then TFUNC would have to be modified by the user to include the factor $1 + bT$. The material properties read into the program would then be zero temperature properties.

IV. USER INFORMATION

This chapter describes the manner in which problems should be formulated. Also the procedures used in inputting data into the program are included.

A. INPUT DATA

Data for each problem processed by this program is assembled in groups. The first group includes the titles and all the control parameters. The second group is the material properties. The third group is the time sequencing information. The fourth group is nodal point information. The fifth group is element information. The sixth group is boundary conditions. After each data deck there must be three blank cards.

B. CONTROL PARAMETERS

The following control parameters are read into the program.

NUMNP	number of nodal point
NUMEL	number of elements
NUMCBC	number of convection boundary conditions
NUMMAT	number of materials
NSEQ	number of time sequences
KAT	type of geometry, 0 implies plane geometry
INTER	output print interval
UNIT	units system
KPUNCH	spacial temperature output
JUMP	generated temperature of flux boundary conditions
KEY	generated convection boundary conditions
TO	initial temperature

Also included with the control data are two labels. TITLE (18) is an arbitrary label that may be used to identify individual problems. It is 72 spaces long. The second label is UN(12). UN(12) is an array of four unit labels. Each has 12 spaces. The units must be printed in the following order: length, mass, time, and temperature. These labels must be left justified.

C. MATERIAL PROPERTIES

The following material properties are read into the program.

XCOND	k_{xx}
YCOND	k_{yy}
XYCOND	k_{xy}
SPHT	ρc
HX	heat generation

Material properties are read in with one material to a card. If the material is isotropic $k_{xx} = k_{yy} = 0$. If the material is anisotropic the three values will not be the same. Each material is given a number according to the order it is read into the program.

If temperature dependent material properties are used both logical variables must be true. Also both JUMP and KEY must be some number other than zero and FCBC,FTBC, and FFBC must be one.

D. TIME SEQUENCING INFORMATION

The following time sequencing information is read into the program.

LTAG	temperature dependent properties (true or false)
MTAG	time dependent boundary conditions (true or false)
ITAG	number of time steps
TAG	time increment

The time sequencing information controls the operation of the program through the use of the logical variables LTAG and MTAG but it can also be used to slow or speed time during the problem. An example would be two sequences with the first solving for the temperature 60 times during the first hour of the problem (i.e., every minute). The second time sequence would solve for the temperature 6 times in the second hour of problem time (i.e., every 10 minutes).

E. NODAL POINT CONSTRUCTION

The following nodal point information is read into the program for nodal point data.

N	number of the nodal point
KODE	1 indicates a temperature boundary condition
X	x coordinate of the nodal point
Y	y coordinate of the nodal point
T	initial temperature of the nodal point

The data for each nodal point is put on one card.

Earlier it was mentioned that the program could generate nodal points. This feature has a few limitations that must be observed or incorrect information will result. A nodal point whose KODE is 1 cannot be generated. Nodal points must be generated along straight lines. The initial temperature of the generated nodal points must be the same and equal to TO, the initial temperature. To generate nodal points simply leave out the data cards between two nodal points on a straight line and the program will generate the information for the nodal points in between.

In constructing the mesh of nodal points for a problem it is sometimes impossible to divide the area to be analyzed into uniform rectangular quadrilaterals. At times it will not be desirable to do so. Consider figure 4-1. In preparing a nodal mesh for this configuration the first step would be to assume that by symmetry there is no heat flux across the center lines. This reduces the figure to one of its quadrants as in

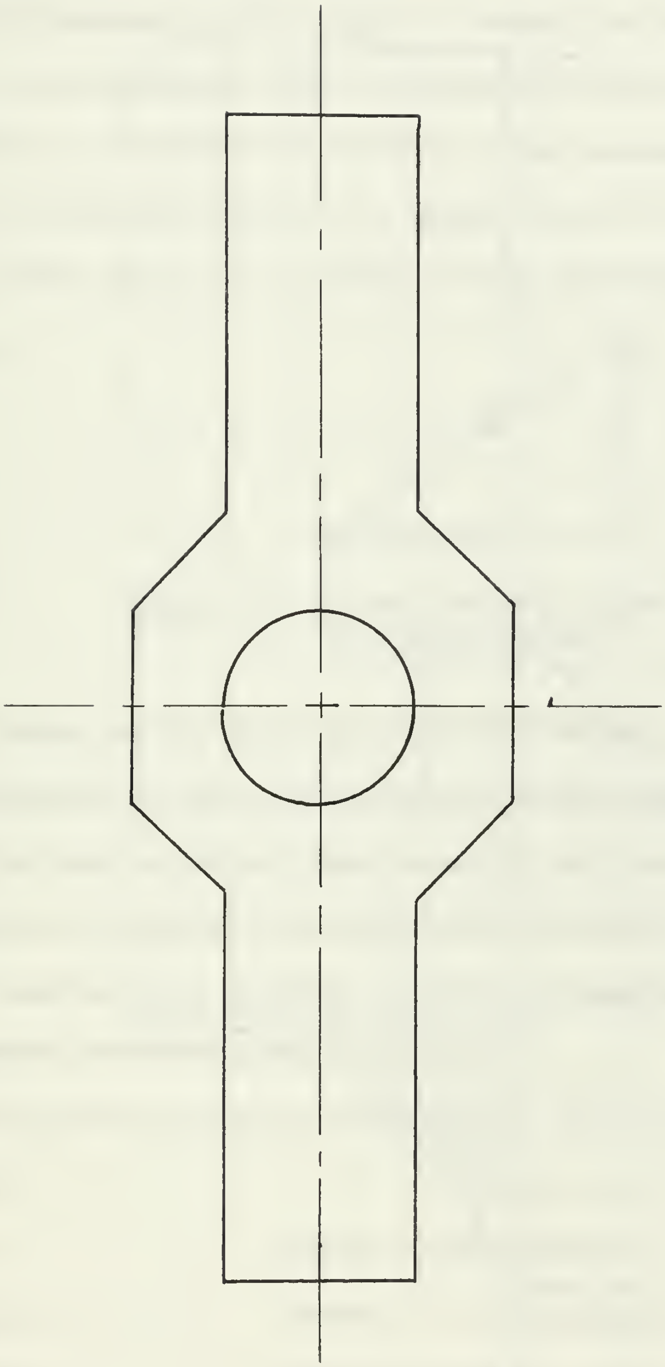

Figure 4-1 Irregularly Bounded Configuration

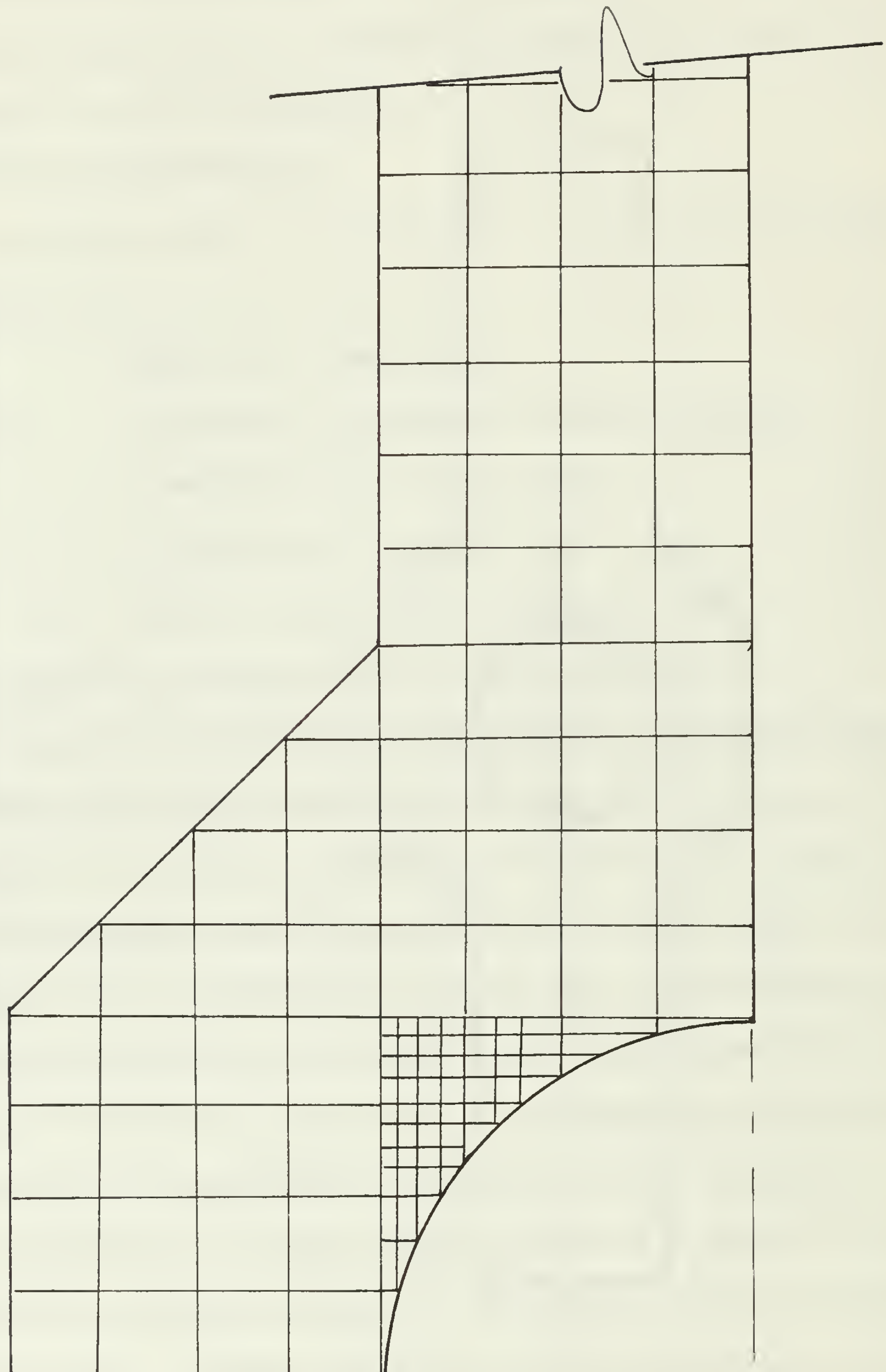

Figure 4-2 Reduced Irregularly Bounded Configuration

figure 4-2. The size of the quadrilateral elements would vary depending upon how interested the user was in a certain area of the figure. In figure 4-2 the area around the curve is being more closely examined but the mesh size must also be small to enable the user to approximate any curved surface using a series of straight line segments. It is almost impossible to divide an area with a curved boundary without encountering a right triangular element. To convert this to a quadrilateral element place a nodal point in the middle of the hypotenuse.

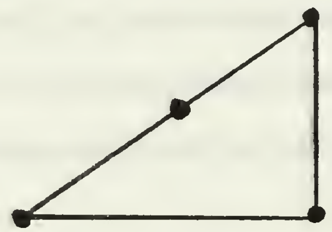

Figure 4-3 Conversion of a Right-Triangle to a Quadrilateral

In numbering the nodal points it must be kept in mind that the maximum difference in nodal point numbers in one element is 31. This is a limitation set by the half-band width of the computer program. It is convenient to increase the numbering of the nodal points and the elements in the same pattern, although it is not a requirement.

F. ELEMENT CONSTRUCTION

The following element information is read into the program for element data.

N	number of the element
IX(N,K) K = 1 - 4	numbers of the element's nodal points
IX(N,5)	material type
HTGEN	element heat generation

The data for each element must be placed on one card. The nodal point numbers are placed in clockwise order starting with the number in the upper left hand corner. Elements may be generated by the program but there are some limitations. Only elements of the same material type can be generated at one time. The heat generation must be the same for the elements generated together. Elements must be generated in rows or columns. To generate elements simply leave out those element cards that you wish generated.

G. BOUNDARY CONDITIONS

There are two types of boundary conditions that may be read into the program, convection and temperature or flux. (An adiabatic boundary condition is a flux boundary condition with the flux equal to zero.) The following information is read into the program for convection boundary conditions.

IB	nodal point number
JB	nodal point number
HB	convective heat transfer coefficient
TEMPB	ambient fluid temperature

The data for each portion of the convection boundary is put one one card. If time varying ambient temperature is to be generated by the program the control parameter KEY must be some number other than zero, say one. If the time varying ambient temperature is not generated by the program new data for each segment of the convective boundary must be read into the program with each new time step. In this case KEY is zero.

The following information is read into the program for temperature and flux boundary conditions.

NN	nodal point number
TQ	temperature or flux

The data for the temperature or flux boundary condition at each nodal point is put on one card. If time varying temperature or flux is to be generated by the program the control parameter JUMP must be some number other than zero. If time varying temperature or flux is not generated by the program new data for each boundary nodal point must be read into the program with each new time step. In this case JUMP is zero. Both temperature and flux boundary conditions must be introduced into the program in the same manner. For both convection and temperature or flux boundary conditions, if time variations are used the logical variable MTAG must be true. Also, regardless of the position of the last nodal point, a boundary condition card must be read into the program for it. If the last nodal point is not a boundary point then the data card consists of the nodal point number and a value of zero in the temperature or flux field.

H. UNITS

The units used for each problem must be consistant with the input data or the labels on the output will be incorrect. Presently there are six different units systems available. Each system consists of a length, mass, time, and temperature. Each system has a number which must be read in as a control parameter (UNIT). The systems are:

1. feet, pound mass, hours, ^{o}F
2. inches, pound mass, seconds, ^{o}F
3. feet, pound mass, seconds, ^{o}F
4. meters, kilograms, seconds, ^{o}C
5. centimeters, grams, seconds, ^{o}C
6. feet, pound mass, minutes, ^{o}F

If the user does not prefer the units available he can use another system in his input and change the labels or just ignore the labels altogether. The system of units will depend greatly upon the configuration of the problem and the temperature differences involved.

I. ERROR EXITS

There are three error exits in the program. The first occurs if a non-positive number of nodal points is read into the program. This exit is used to stop the program. The other two occur in the nodal point and element information reading sections. If these statements are executed then the execution of the program is terminated. If either of the last two exits are executed a statement will be printed to indicate which one it was. Usually termination means a bad data card or that the user was trying to generate points beyond the capability of the program.

J. COORDINATES

The cartesian system of coordinates x and y are used for problems involving plane geometry. When axisymmetric problems are considered the x,θ,z system of polar coordinates is used. The analysis of axisymmetric problems is conducted in the x,z plane.

K. TEMPERATURE OUTPUT

The temperature output can be modified in two ways. The number of time increments printed out can be limited by using the control parameter INTER. If the program is directed to solve for the temperature every minute for 100 minutes and INTER is equal to 25, temperature information will only be printed for 25, 50, 75, and 100 minutes. The second manner in which the output may be modified is at the end of each time sequence the x,y coordinates of each nodal point and the temperature are printed if the control parameter KPUNCH equals one.

L. INPUT DATA PREPARATION

The following sections describe the manner in which the input data is to be placed on data cards. The data is prepared in groups and the final data deck is assembled by putting the data groups together in order of their listing below.

1. Control Information

(a) Title Card (18A4): 72 arbitrary spaces for an alphameric problem label punched in columns 1-72

(b) Units Card (12A4): Alphameric unit labels

Columns	1-12	length
	13-24	mass
	25-36	time
	37-48	temperature

Labels should be left justified.

(c) Control Information Card (11I5, F10.0)

Columns	1-5	NUMNP	number of nodal points
	6-10	NUMEL	number of elements
	11-15	NUMCBC	number of convection boundary conditions
	16-20	NUMMAT	number of materials
	21-25	NSEQ	number of time sequences
	26-30	KAT	type of geometry
	31-35	INTER	print interval
	36-40	UNIT	units system (integer)
	41-45	KPUNCH	spacial distribution
	46-50	JUMP	temperature conditions
	51-55	KEY	convection conditions
	56-65	TO	initial temperature

"I" formats must be right justified.

2. Material Properties

(a) Material Properties Card (5D10.3)

Columns	1-10	XCOND	k_{xx}

	11-20	YCOND	k_{yy}
	21-30	XYCOND	k_{xy}
	31-40	SPHT	ρc
	41-50	HX	heat generation

One card for each material.

3. Time Sequencing Information

(a) Time Sequencing Card (2L5,I10,D10.3):

Columns	1-5	LTAG	temperature dependent properties (true or false)
	6-10	MTAG	time dependent boundary conditions (true or false)
	11-20	ITAG	number of time steps
	21-21	TAG	time increment

One card for each time sequence.

4. Nodal Point Information

(a) Nodal Point Card (2I5,3F10.0):

Columns	1-5	N	nodal point number
	6-10	KODE	temperature boundary condition
	11-20	X	x coordinate
	21-30	Y	y coordinate
	31-40	T	initial temperature

One card for each nodal point not generated by the program.

5. Element Information

(a) Element Card (6I5,D10.3):

Columns	1-5	N	element number
	6-10	IX(N,1)	nodal point number

11-15	IX(N,2)	nodal point number
16-20	IX(N,3)	nodal point number
21-25	IX(N,4)	nodal point number
26-30	IX(N,5)	material number
31-40	HTGEN	element heat generation

One card for each element not generated. Nodal points are numbered clockwise starting in the upper left hand corner.

6. Convection Boundary Conditions

(a) Convection Boundary Card (2I5,D10.3,F10.0)

Columns	1-5	IB	nodal point number
	6-10	JB	nodal point number
	11-20	HB	convective heat transfer coefficient
	21-30	TEMPB	ambient fluid temperature

One card for each convection boundary condition.

7. Temperature and Flux Boundary Conditions

(a) Temperature or Flux Boundary Card (I10,D10.3)

Columns	1-10	NN	nodal point number
	11-20	TQ	temperature or flux

One card for each temperature or flux boundary point. A temperature or flux boundary card for the last nodal point must be read even if it isn't a boundary point.

8. Multiple Problems

The above cards comprise a data deck for one problem. Multiple problems may be solved by putting their data decks together. The only limitation on this procedure is computer time. It is important that there be three blank cards after the last problem data set in the data deck. This is to terminate operation of the program.

RECOMMENDATIONS

Since the number of problem types that can be solved by this program is limited mostly by the imagination of the user it is highly recommended that this engineering tool should be brought to the attention of those students studying in the area of unsteady heat conduction. Only through complete familiarization with the program can the user make use of its great potential to solve problems involving complex configurations and boundary conditions.

Recommendations for further study in this area are:

1. Expansion of the finite element method to three dimensional configurations.

2. Revision of the boundary condition segments of the program to make use of random access disks for temporary storage.

3. Construction of a supplementary program that will punch input data cards for complex time dependent boundary conditions.

4. An isothermal contour plotting subroutine capable of being used in conjunction with the present NPS Computer Center subroutine DRAW.

5. Use of the program in conjunction with the NPS Computer Center IBM Optical Display Unit, in order to observe transient heat conduction differences after minor alterations in mesh configuration.

6. Use of the program in conjunction with existing finite element programs in the area of stress analysis to obtain thermal stresses.

APPENDIX A

In this section some solutions to different configurations will be presented as examples of possible problem types that may be solved with this program. Also the results of these examples are compared with analytic solutions of the same configurations. In order to avoid confusion the same material nickel (99.9%) and the same initial temperature, $100^{o}F$, are used in each problem.

The material properties of nickel are:

$$\rho = 556 \text{ lbm/ft}^3$$

$$c = .1065 \text{ BTU/lbm-}^{o}\text{F}$$

$$\alpha_T = .882 \text{ ft}^2\text{/hr}$$

Example 1.

This is a problem in one dimensional heat conduction. It could also be considered a problem of heat conduction in a semi-infinite plate by assuming that lines parallel to the direction of heat flux are adiabatic boundaries. Because of this assumption we only need one row of elements across the plate which is one foot wide (see figure A-1). The top and bottom sides of the elements are adiabatic boundaries. Also the end boundary conditions are adiabatic on the left and are constant $0^{o}F$ on the right. The nodal points are numbered from left to right, starting with the top row. The elements are also numbered from left to right. The computer solution provides answers at distinct time intervals while most

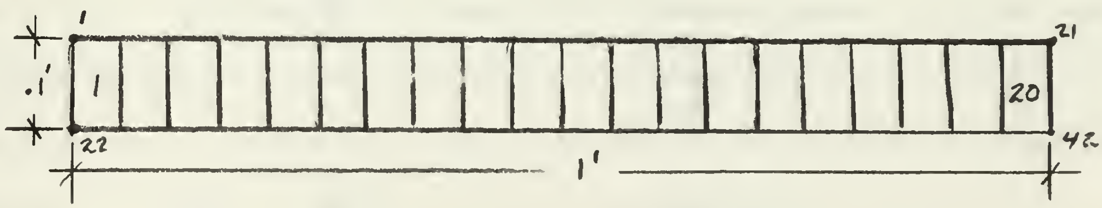

Figure A-1 One Dimensional Heat Conduction analytic solutions are given in terms of the dimensionless time parameter Fourier number, N_{FO}. The Fourier number, N_{FO}, equals $\alpha_T t/L^2$, where α_T is the thermal diffusivity of the material. The comparisons will be made at various Fourier numbers. The analytic solution was obtained from page 98 of reference [2].

Nodal Point	N_{FO}	Carslaw and Jaeger	Finite Element
1	.04	100	99.6
5	.04	99	99.0
9	.04	96	95.9
13	.04	85	84.7
17	.04	54	54.5
1	.1	95	94.8
5	.1	93	91.5
9	.1	82	82.5
13	.1	63	63.6
17	.1	35	35.1
1	.4	48	48.1
5	.4	45	45.8
9	.4	38	38.9
13	.4	28	28.3
17	.4	15	14.9

Example 2.

This example has the same mesh configuration and boundary conditions as the first example. The initial temperature distribution is changed to the form $T = 100 - 50x\,^{o}F$. The analytic solution was obtained from page 98 of reference [2].

Nodal Point	N_{FO}	Carslaw and Jaeger	Finite Element
1	.01	95	94.0
5	.01	90	90.0
9	.01	80	79.8
13	.01	70	68.0
17	.01	60	53.0
1	.04	99	89.0
5	.04	86	85.6
9	.04	77	76.9

13	.04	62	62.1
17	.04	36	37.0
1	.1	80	80.0
5	.1	76	76.0
9	.1	67	66.3
13	.1	50	49.7
17	.1	28	27.0

Example 3.

This example is a solution for two-dimensional heat conduction. The configuration is a square plate with an initial temperature of $100^{o}F$ and boundary conditions of $0^{o}F$ on all sides. Using symmetry the user need only examine one fourth of the plate, i.e. one corner (see figure A-2).

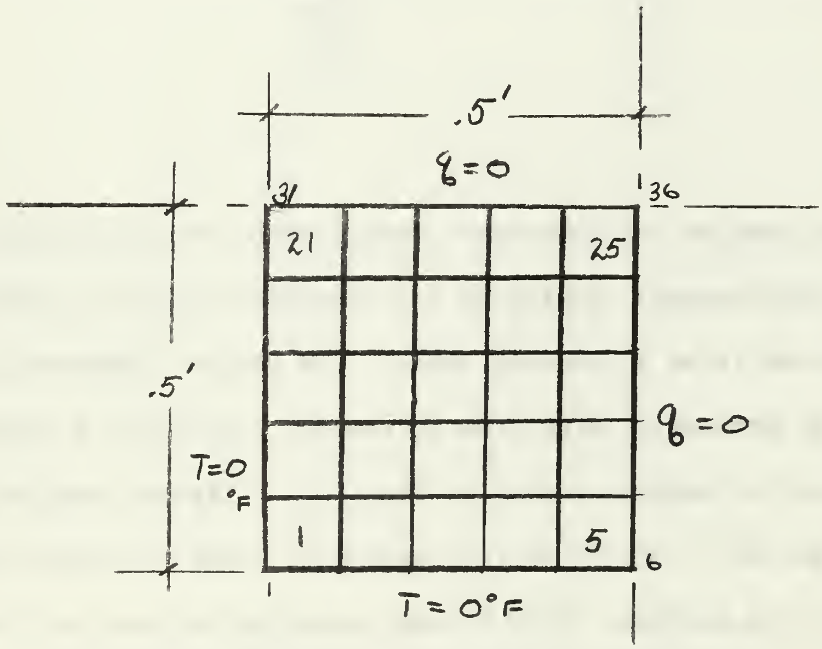

Figure A-2 Two Dimensional Heat Conduction

The nodal points are numbered from left to right and from the bottom row to the top. The elements are numbered in the same manner. In this case the problem consists of a plate with two sides at $0^{o}F$ and the other two insulated for an adiabatic boundary condition. The analytic solution is obtained by using the solution for one dimensional heat conduction in a semi-infinite plate and the method of multiplicative superposition.

The analytic solution was obtained from page 118 of reference [3].

Nodal Point	N_{FO}	Chapman	Finite Element
8	.01	27.0	30.0
15	.01	70.6	70.6
22	.01	92.0	90.5
29	.01	97.0	97.3
36	.01	100.0	98.8
8	.05	5.75	6.29
15	.05	21.2	22.2
22	.05	40.4	40.8
29	.05	54.0	55.1
36	.05	59.3	60.4
8	.1	1.96	2.23
15	.1	7.3	8.1
22	.1	14.4	15.2
29	.1	19.4	21.1
39	.1	22.1	23.3

Example 4.

In this problem the same mesh configuration as the previous example is used. The boundary conditions are changed so that the sides are adiabatic, the top is at a constant 100°F. The initial temperature is 100°F. By taking an extremely large time increment, that is of a magnitude greater than the maximum number of significant figures available in the computer, say 10^{17}, the first time step will yield the steady state solution to the problem. This is made possible because the time dependent contributions to the coefficient matrix and the forcing vector are multiplied by the inverse of the time increment which in this case is essentially zero (10^{-17}). The solution to this problem is a linear temperature distribution and the results below were obtained by using a time increment of 10^{17} minutes.

Nodal Point	Finite Element
3	0.0
9	20.0

15	40.0
21	60.0
27	80.0
33	100.0

APPENDIX B

A	coefficient matrix
ACALTR	dummy logical variable
B	effective forcing vector
CONDI	k_x in an element
CONDJ	k_y in an element
CONDJ	k_{xy} in an element
H	convection boundary layer heat transfer coefficient
HX	heat generation in a material
INTER	output print interval
ITAG	number of time steps
IX	k = 1 - 4 nodal point numbers
	k = 5 material number
JUMP	temperature and flux boundary generation
KAT	geometry type
KEY	convection boundary generation
KODE	temperature or flux boundary parameter
KPUNCH	spacial temperature distribution parameter
LPRINT	print parameter
LTAG	logical variable
MBAND	half band width
MTAG	logical variable
MTYPE	material number
NSEQ	number of time sequences
NUMCBC	number of convection boundary conditions
NUMEL	number of elements
NUMMAT	number of materials

NUMNP	number of nodal points
SPHT	the product ρc
T	nodal temperature
TAG	time increment
TEMP	ambient fluid temperature
TIME	time variable
TITLE	identification title
TMEAN	mean temperature in an element
TO	initial temperature
UN	unit labels
UNIT	unit system parameter
X	nodal point x coordinate
XCOND	k_x in a material
XMEAN	mean x coordinate of an element
XYCOND	k_{xy} in a material
Y	nodal point y coordinate
YCOND	k_y in a material
YMEAN	mean y coordinate of an element

APPENDIX C

*Title and Labels

```
     THESIS CHECK PROBLEM HEAT CONDUCTION IN A RECTANGLE
FEET        POUND MASS  MINUTES      FAHRENHEIT
```

*Control Information

```
36   25    0    1    1    0    1    6    1    0    0100.0
```

*Material Properties

```
+8.660D-01+8.660D-01+0.000D+00+5.930D+01+0.000D+00
```

*Time Sequencing Information

```
FALSEFALSE          250+1.000D-01
```

*Nodal Point Information

```
 1     10.0      0.0      100.0
 2     10.1      0.0      100.0
 3     10.2      0.0      100.0
 4     10.3      0.0      100.0
 5     10.4      0.0      100.0
 6     10.5      0.0      100.0
 7     10.0      0.1      100.0
12     00.5      0.1      100.0
13     10.0      0.2      100.0
18     00.5      0.2      100.0
19     10.0      0.3      100.0
24     00.5      0.3      100.0
25     10.0      0.4      100.0
30     00.5      0.4      100.0
31     10.0      0.5      100.0
36     00.5      0.5      100.0
```

*Element Information

```
 1    7    8    2    1   1+0.000D+00
 5   11   12    6    5   1+0.000D+00
 6   13   14    8    7   1+0.000D+00
10   17   18   12   11   1+0.000D+00
11   19   20   14   13   1+0.000D+00
15   23   24   18   17   1+0.000D+00
16   25   26   20   19   1+0.000D+00
20   29   30   24   23   1+0.000D+00
21   31   32   26   25   1+0.000D+00
25   35   36   30   29   1+0.000D+00
```

*Boundary Conditions

```
1+0.000D+00
2+0.000D+00
3+0.000D+00
4+0.000D+00
5+0.000D+00
```

```
 6+0.0000+00
 7+0.0000+00
13+0.0000+00
19+0.0000+00
25+0.0000+00
31+0.0000+00
36+0.0000+00
```

NOTE: Starred Lines are not Data Cards.

```
C
C     **********************************************************************
C*****      TRANSIENT HEAT CONDUCTION ANALYSIS FOR PLANE AND AXISYMMETRIC
C*****      SOLIDS WRITTEN BY R.E. NICKELL, LUMPED HEAT CAPACITY, BASED ON
C*****      A VARIATIONAL PRINCIPLE DERIVED BY M.E. GURTIN, PROGRAM MODIFIED BY
C*****      A.B. CHALOUPKA
C     **********************************************************************
C
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/SYMARG/ NUMNP,MBAND,A(500,32),B(500)
      COMMON/ELPRCP/
     1XCOND(25),YCOND(25),XYCOND(25),SPHT(25),HX(25),HTGEN(490),KAT
      COMMON/NPDATA/ X(500),Y(500),T(500)
      COMMON/ELADD/ TC(5,5),QP(5),HC(5)
      DIMENSION
     1TITLE(18),LTAG(10),MTAG(10),TAG(10),D(500),KODE(500),ITAG(500)
      LOGICAL LTAG,MTAG,ACALTR,BCALTR
      DIMENSION Q(500),IX(490,5),UN(12),TFX(500)
      DIMENSION TEMPB(500),HB(500),TQ(500),IB(500),JB(500)
      INTEGER UNIT
      DATA ITAPE/9/
C
C     **********************************************************************
C*****     READ AND PRINT CONTROL INFORMATION
C     **********************************************************************
C
   10 READ (5,1000) (TITLE(J),J=1,18),(UN(K),K=1,12),NUMNP,NUMEL,NUMCBC,
     1NUMMAT,NSEQ,KAT,INTER,UNIT,KPUNCH,JUMP,KEY,TO
C*****     (KAT.EQ.0) PLANE GEOMETRY  -----  (KAT.NE.0) AXISYMMETRIC GEOMETRY
      IF (NUMNP.LE.0) GO TO 900
      WRITE (6,2000) (TITLE(J),J=1,18)
      WRITE (6,2003) (UN(J),J=1,12)
      IF (KAT.EQ.0) WRITE (6,2001)
      IF (KAT.NE.0) WRITE (6,2002)
      WRITE (6,1999) NUMNP,NUMEL,NUMCBC,NUMMAT,NSEQ,INTER,TO,(UN(J),J=10
     1,12)
C
C     **********************************************************************
C*****     READ AND PRINT OF MATERIAL PROPERTIES
C     **********************************************************************
C
      WRITE (6,2004)
      IF (KAT.EQ.0) WRITE (6,1997)
      IF (KAT.NE.0) WRITE (6,1998)
      DO 25 M=1,NUMMAT
      READ (5,1005) XCOND(M),YCOND(M),XYCOND(M),SPHT(M),HX(M)
      WRITE (6,2005) M,XCOND(M),YCOND(M),XYCOND(M),SPHT(M),HX(M)
      IF (UNIT.EQ.1) WRITE (6,2006)
```

```
      IF (UNIT.EQ.2) WRITE (6,2007)
      IF (UNIT.EQ.3) WRITE (6,2008)
      IF (UNIT.EQ.4) WRITE (6,2011)
      IF (UNIT.EQ.5) WRITE (6,2012)
      IF (UNIT.EQ.6) WRITE (6,2016)
   25 CONTINUE
C
C     ******************************************************************
C*****     READ TIME SEQUENCING INFORMATICN
C     ******************************************************************
C
      WRITE (6,1995)
      DO 50 N=1,NSEQ
      READ (5,1010) LTAG(N),MTAG(N),ITAG(N),TAG(N)
      WRITE (6,2010) N,LTAG(N),MTAG(N),ITAG(N),TAG(N),(UN(K),K=7,9)
   50 CONTINUE
C
C     ******************************************************************
C*****     READ OR GENERATE NODAL POINT INFORMATION
C     ******************************************************************
C
      WRITE (6,1996)
      IF (KAT.EQ.0) WRITE (6,2013)
      IF (KAT.NE.0) WRITE (6,2014)
      L=1
   60 READ (5,1015) N,KODE(N),X(N),Y(N),T(N)
      DIFF=N-L+1
      IF (N-L) 65,80,70
   70 DX=(X(N)-X(L-1))/DIFF
      DY=(Y(N)-Y(L-1))/DIFF
   75 KODE(L)=0
      X(L)=X(L-1)+DX
      Y(L)=Y(L-1)+DY
      T(L)=TO
   80 WRITE (6,2015) L,KODE(L),X(L),(UN(J),J=1,3),Y(L),(UN(K),K=1,3),
     1T(L),(UN(M),M=10,12)
      L=L+1
      IF (N-L) 90,80,75
   90 IF (NUMNP+1-L) 100,100,60
   65 WRITE (6,2100) N
      GO TO 900
  100 CONTINUE
C
C     ******************************************************************
C*****      READ OR GENERATE ELEMENT DESCRIPTION, DETERMINE HALF-BAND WIDTH
C     ******************************************************************
C
      WRITE (6,2018)
```

```
      MBAND=0
      N=0
      DO 200 M=1,NUMEL
      IF (N.LT.M) READ (5,1020) N,(IX(N,K),K=1,5),HTGEN(N)
      IF (N-M) 210,215,220
  210 WRITE (6,2200) M
      GO TO 900
  220 IX(M,1)=IX(M-1,1)+1
      IX(M,2)=IX(M-1,2)+1
      IX(M,3)=IX(M-1,3)+1
      IX(M,4)=IX(M-1,4)+1
      IX(M,5)=IX(M-1,5)
      HTGEN(M)=HTGEN(M-1)
  215 WRITE (6,2020) M,(IX(M,K),K=1,5),HTGEN(M)
      DO 230 I=1,4
      DO 230 J=1,4
      MM=IABS(IX(M,I)-IX(M,J))
      IF (MM+1.GT.32) GO TO 210
      IF (MM+1.GT.MBAND) MBAND=MM+1
  230 CONTINUE
  200 CONTINUE
C
C     ****************************************************************
C*****     BEGIN TRANSIENT CALCULATIONS
C     ****************************************************************
C
      TIME=0.0D+00
      LPRINT=0
      LIMU=0
      DO 800 NS=1,NSEQ
      ACALTR=LTAG(NS)
      BCALTR=MTAG(NS)
      LIML=LIMU+1
      LIMU=LIMU+ITAG(NS)
      DT=2.*TAG(NS)
      DT2=2./DT
C
C     ****************************************************************
C*****     EVALUATE FOR EACH TIME STEP IN THE SEQUENCE
C     ****************************************************************
C
      DO 600 LNDT=LIML,LIMU
      IF (LNDT.GT.LIML.AND..NOT.ACALTR) GO TO 605
      IF(NS.GT.1.AND..NOT.ACALTR) GO TO 605
      DO 250 N=1,NUMNP
      D(N)=0.0D+00
      Q(N)=0.0D+00
      TFX(N)=0.0D+00
```

```
      DO 250 M=1,MBAND
  250 A(N,M)=0.
C
C     ****************************************************************
C*****     FORM HEAT CAPACITY AND CONDUCTIVITY MATRICES
C     ****************************************************************
C
      DO 275 N=1,NUMEL
      II=IX(N,1)
      JJ=IX(N,2)
      KK=IX(N,3)
      LL=IX(N,4)
      MTYPE=IX(N,5)
      CALL FORM (N,II,JJ,KK,LL,MTYPE)
      DO 280 L=1,4
      I=IX(N,L)
      D(I)=D(I)+HC(L)
      Q(I)=Q(I)+CP(L)
      DO 280 M=1,4
      J=IX(N,M)-I+1
      IF (J.LE.0) GO TO 280
      A(I,J)=A(I,J)+TC(L,M)
  280 CONTINUE
  275 CONTINUE
C
C     ****************************************************************
C*****     WRITE TIME INDEPENDENT INFORMATION ON TAPE
C     ****************************************************************
C
      IF (ACALTR) GO TO 610
      WRITE (ITAPE) ((A(I,J),J=1,MBAND),D(I),Q(I),I=1,NUMNP)
      REWIND ITAPE
  605 IF (LNDT.EQ.LIML.OR..NOT.BCALTR) GO TO 610
      IF(NS.GT.1.AND..NOT.BCALTR) GO TO 610
      READ (ITAPE) ((A(I,J),J=1,MBAND),D(I),Q(I),I=1,NUMNP)
      REWIND ITAPE
  610 IF (LNDT.GT.LIML.AND..NOT.BCALTR) GO TO 615
      IF(NS.GT.1.AND..NOT.BCALTR) GO TO 615
C
C     ****************************************************************
C*****     CONVECTION BOUNDARY CONDITIONS
C     ****************************************************************
C
      IF (NUMCBC.EQ.0) GO TO 300
      WRITE (6,2024)
      DO 310 N=1,NUMCBC
      IF(LNDT.GT.1.AND.KEY.NE.0) GO TO 301
      READ (5,1025) IB(N),JB(N),HB(N),TEMPB(N)
```

```
      WRITE (6,2025) IB(N),JB(N),HB(N),TEMPB(N)
  301 I=IB(N)
      J=JB(N)
      H=HB(N)
      TEMP=TEMPB(N)
      TEMP=TEMP*FCBC(TIME)
      XL=DSQRT((X(J)-X(I))**2+(Y(J)-Y(I))**2)
      IF (KAT.NE.0) XL=.5*XL*(X(I)+X(J))
      TEMP=.5*TEMP*H*XL
      H=H*XL/6.
      Q(I)=Q(I)+TEMP
      Q(J)=Q(J)+TEMP
      A(I,1)=A(I,1)+2.*H
      A(J,1)=A(J,1)+2.*H
      K=J-I+1
      IF (K.GT.0) A(I,K)=A(I,K)+H
      K=I-J+1
      IF (K.GT.0) A(J,K)=A(J,K)+H
  310 CONTINUE
  300 CONTINUE
C
C     ***********************************************************************
C*****       TEMPERATURE AND FLUX BOUNDARY CONDITIONS
C     ***********************************************************************
C
      IF (LNDT.GT.1.AND.JUMP.NE.0) GC TO 401
      DO 401 N=1,NUMNP
  601 TQ(N)=0.
  325 READ (5,1035) NN,TQ(NN)
      IF (KCDE(NN).EQ.1) TFX(NN)=TQ(NN)
      IF (NN.LT.NUMNP) GO TO 325
  401 DO 400 N=1,NUMNP
      IF (KODE(N).EQ.0) TQ(N)=TQ(N)*FFBC(TIME)
      IF (KODE(N).EQ.0) GO TO 335
      DO 350 M=2,MBAND
      IF (KCDE(N).EQ.1) TQ(N)=TQ(N)*FTBC(TIME)
      K=N-M+1
      IF (K.LE.0.CR.KODE(K).NE.0) GC TO 345
      Q(K)=Q(K)-A(K,M)*(TQ(N)-TO)
      A(K,M)=0.
  345 K=N+M-1
      IF (K.GT.NUMNP.CR.KODE(K).NE.0) GO TO 350
      Q(K)=Q(K)-A(N,M)*(TQ(N)-TO)
      A(N,M)=0.
  350 CONTINUE
      A(N,1)=1.-DT2*D(N)
      Q(N)=TQ(N)-TO-CT2*D(N)*(T(N)-TO)
      GO TO 320
```

```
  335 Q(N)=Q(N)+TC(N)                                                          177
C
C     ****************************************************************************
C*****     FORM EFFECTIVE COEFFEICIENT MATRIX FOR TIME INCREMENT
C     ****************************************************************************
C
  320 A(N,1)=A(N,1)+DT2*D(N)                                                   177
  400 CONTINUE                                                                 178
C
C     ****************************************************************************
C*****     PUT MATRIX IN UPPER TRIANGULAR FORM
C     ****************************************************************************
C
      CALL SYMSCL (1)                                                          179
C
C     ****************************************************************************
C*****     CALCULATE TEMPERATURE AT THE MIDDLE OF TIME INCREMENT
C     ****************************************************************************
C
189   CALL SYMSCL (1)
  615 TIME=TIME+.5*DT                                                          180
      LPRINT=LPRINT+1                                                          181
C
C     ****************************************************************************
C*****     CALCULATE EFFECTIVE FORCING VECTOR
C     ****************************************************************************
C
      DO 450 N=1,NUMNP                                                         182
      B(N)=Q(N)+DT2*D(N)*(T(N)-TO)                                             183
  450 CONTINUE                                                                 184
C
C     ****************************************************************************
C*****     CALL EQUATION SOLVER
C     ****************************************************************************
C
      CALL SYMSCL (2)                                                          185
C
C     ****************************************************************************
C*****     CALCULATE ABSOLUTE TEMPERATURES
C     ****************************************************************************
C
      DO 500 N=1,NUMNP                                                         186
      IF (KODE(N).EQ.1) B(N)=(TFX(N)-TO)                                       187
  500 T(N)=B(N)+TO                                                             188
C
C     ****************************************************************************
C*****     PRINT TEMPERATURES
C     ****************************************************************************
```

```
C
      IF (LPRINT.LT.INTER) GO TO 600
      WRITE (6,2050)TIME,(UN(J),J=7,9   ),(N,T(N),N=1,NUMNP)
      LPRINT=0
  600 CONTINUE
  800 CONTINUE
      IF (KPUNCH.NE.0) WRITE (6,2059)
      IF (KPUNCH.NE.0) WRITE (6,2060) (N,X(N),(UN(J),J=1,3),Y(N),(UN(J),
     1J=1,3),T(N),(UN(J),J=10,12),N=1,NUMNP)
      WRITE (6,2300)
      GO TO 10
C
C     ********************************************************************
C*****      FORMAT STATEMENTS
C     ********************************************************************
C
 1000 FORMAT (18A4/12A4/11I5,F10.0)
 1005 FORMAT (8D10.3)
 1010 FORMAT (2L5,I10,D10.3)
 1015 FORMAT (2I5,3F10.0)
 1030 FORMAT (6I5,D10.3)
 1025 FORMAT(2I5,D10.3,F10.0)
 1035 FORMAT (I10,D10.3)
 1995 FORMAT (1H0,///,45X,'TIME SEQUENCING INFORMATION',/,45X,'**** ****
     1****** ***********',//,1H0,//,' TIME SEQUENCE    TEMP. DEP. PROP.
     2  TIME DEP.B.C.   NO. OF TIME STEPS       TIME INCREMENT',//)
 1996 FORMAT (1H0,///,47X,'NODAL POINT INFORMATION',/,
     147X,'***** ***** ***********',//)
 1997 FORMAT (1H0,'MATERIAL',12X,'KXX',24X,'KYY',24X,'KXY',24X,'C',14X,'
     1Q',//)
 1998 FORMAT (1H0,'MATERIAL',12X,'KRR',24X,'KZZ',24X,'KRZ',24X,'C',14X,'
     1Q',//)
 1999 FORMAT (1H0,'     NUMBER OF NODAL POINTS--',I4,//,
     1'         NUMBER OF ELEMENTS------',I4,//,
     2'         NUMBER OF CONVECTION BC-',I4,//,
     3'         NUMBER OF MATERIALS-----',I4,//,
     4'         NUMBER OF TIME SEQUENCES',I4,//,
     5'         OUTPUT PRINT INTERVAL---',I4,//,
     6'         REFERENCE TEMPERATURE',1PD10.3,2X,3A4)
 2000 FORMAT (1H1,///,23X,18A4,///)
 2001 FORMAT (1H0,51X,'TWO-DIMENSIONAL PLANE BODY',/,
     152X,'*************** ***** ****',//)
 2002 FORMAT (1H0,53X,'AXISYMMETRIC SOLID BODY',/,
     154X,'************ ***** ****',//)
 2003 FORMAT(1H0,35X,'UNITS:',5X,12A4,///)
 2004 FORMAT (1H0,///,49X,'MATERIAL PROPERTIES'/,
     149X,'******** **********',//)
 2005 FORMAT(1H ,I5,7X,1PD12.5,1P3D27.5,1PD20.5)
```

```
2006 FORMAT (1H+,24X,' BTU/HR-FT-F ',14X,' BTU/HR-FT-F ',14X,' BTU/FT-F
    1 ')
2007 FORMAT(1H+,24X,' BTU/SEC-IN-F ',13X,' BTU/SEC-IN-F ',13X,' BTU/SEC
    1-IN-F ')
2008 FORMAT(1H+,24X,' BTU/SEC-FT-F ',13X,' BTU/SEC-FT-F ',13X,' BTU/SEC
    1-FT-F ')
2010 FORMAT (6X,I2,16X,L1,20X,L1,14X,I4,11X,1PD12.4,2X,3A4)
2011 FORMAT (1H+,24X,' CAL/SEC-M-C ',14X,' CAL/SEC-M-C ',14X,' CAL/SEC-
    1M-C')
2012 FORMAT(1H+,24X,' CAL/SEC-CM-C ',13X,' CAL/SEC-CM-C ',13X,' CAL/SEC
    1-CM-C ')
2013 FORMAT (1HO,///,'  N.P. NO.', 2X,'CODE',18X,'X',34X,'Y',35X,'T',/)
2014 FORMAT (1HO,///,'  N.P. NO.', 2X,'CODE',18X,'R',34X,'Z',35X,'T',/)
2015 FORMAT (3X,I4,6X,I1,1PD24.6,2X,3A4,1PD22.6,2X,3A4,1PD22.6,2X,3A4)
2016 FORMAT(1H+,24X,' BTU/MIN-FT-F ',13X,' BTU/MIN-FT-F ',13X,' BTU/MIN
    1-FT-F ')
2018 FORMAT (1HO,///,49X,'ELEMENT DESCRIPTION',/,49X,'******* *********
    1**',//,1HO,'ELEMENT',13X,'I',16X,'J',16X,'K',16X,'L',17X,'MATERIAL
    2',13X,'HEAT GENERATION',//)
2020 FORMAT (I5,4I17,I22,D30.5)
2024 FORMAT (1HO,///,45X,'CONVECTION BOUNDARY CONDITIONS',/,45X,
    1'********** ******** *********',//,1HO,5X,'I',20X,'J',20X,'H',30X,
    2'TEMPERATURE',//)
2025 FORMAT (I7,I21,1PD30.6,1PD30.6,2X,3A4)
2050 FORMAT (1HO,///,6HOTIME:2PD13.5,2X,3A4,///,
    1(I6,1PD14.6,I6,1PD14.6,I6,1PD14.6,I6,1PD14.6,I6,1PD14.6,I6,1PD14.6
    2)/)
2059 FORMAT (1HO,///,' NP.NO.'20X,'X',28X,'Y',28X,'T',//)
2060 FORMAT (1HO,///,(1HO,I5,10X,1PD15.6,2X,3A4,1PD15.6,2X,3A4,1PD15.6,
    12X,3A4))
2100 FORMAT (26HOBAD NODAL POINT CARD NO. I4)
2200 FORMAT (22HOBAD ELEMENT CARD NO. I4)
2300 FORMAT (1HO,///,50X,'END OF PROBLEM',/,
    150X,'*** ** *******')
 900 CONTINUE
     STOP
     END
```

```
      SUBROUTINE FORM (NN,N1,N2,N3,N4,MTYPE)
C
C     ****************************************************************************
C*****     SUBROUTINE TO CALCULATE ELEMENT THERMAL CONDUCTIVITY, HEAT CAPACITY,
C*****     AND HEAT GENERATION
C     ****************************************************************************
C
      IMPLICIT REAL*8 (A-H,C-Z)
      COMMON/ELPRCP/
     1XCOND(25),YCOND(25),XYCOND(25),SPHT(25),HX(25),HTGEN(490),KAT
      COMMON/NPDATA/ X(500),Y(500),T(500)
      COMMON/ELADD/ TC(5,5),QP(5),HC(5)
      DIMENSION E(3,3),MX(3),LM(5)
      DO 100 I=1,5
      QP(I)=0.
      HC(I)=0.
      DO 100 J=1,5
  100 TC(I,J)=0.
C
C     ****************************************************************************
C*****    CALCULATE ELEMENT MATERIAL PROPERTIES
C     ****************************************************************************
C
      TMEAN=(T(N1)+T(N2)+T(N3)+T(N4))/4.
      XMEAN=(X(N1)+X(N2)+X(N3)+X(N4))/4.
      YMEAN=(Y(N1)+Y(N2)+Y(N3)+Y(N4))/4.
      CONDI=XCOND(MTYPE)*TFUNC(MTYPE,TMEAN,1)
      CONDJ=YCOND(MTYPE)*TFUNC(MTYPE,TMEAN,2)
      CONDK=XYCOND(MTYPE)*TFUNC(MTYPE,TMEAN,3)
      QGEN=HX(MTYPE)+HTGEN(NN)
C
C     ****************************************************************************
C*****    FORM QUADRILATERAL FROM FOUR TRIANGLES
C     ****************************************************************************
C
      LM(1)=N1
      LM(2)=N2
      LM(3)=N3
      LM(4)=N4
      LM(5)=N1
      DO 150 K=1,4
      I=LM(K)
      J=LM(K+1)
      IF (I.EQ.J) GO TO 150
      AJ=X(J)-X(I)
      AK=XMEAN-X(I)
      BJ=Y(J)-Y(I)
      BK=YMEAN-Y(I)
```

(MTYPE,TMEAN)

CV=SPHT(MTYPE)*CFUNC

```
      BJK=BJ-BK
      AKJ=AK-AJ
C
C     ***********************************************************************
C*****      FORM CONDUCTIVITY TENSOR FOR ANISOTROPIC MATERIALS
C     ***********************************************************************
C
      E(1,1)=BJK*BJK*CONDI+AKJ*AKJ*CONDJ+2.*BJK*AKJ*CONDK
      E(1,2)=BJK*BK*CONDI-AKJ*AK*CONDJ+CONDK*(AKJ*BK-BJK*AK)
      E(1,3)=-BJK*BJ*CONDI+AKJ*AJ*CONDJ+CONDK*(BJK*AJ-AKJ*BJ)
      E(2,1)=E(1,2)
      E(2,2)=BK*BK*CONDI+AK*AK*CONDJ-2.*AK*BK*CONDK
      E(2,3)=-BK*BJ*CONDI-AJ*AK*CONDJ+CONDK*(AJ*BK+AK*BJ)
      E(3,1)=E(1,3)
      E(3,2)=E(2,3)
      E(3,3)=BJ*BJ*CONDI+AJ*AJ*CONDJ-2.*AJ*BJ*CONDK
      XMUL=1.
      IF (KAT.NE.0) XMUL=XMUL*(X(I)+X(J)+XMEAN)/3.
      AREA=AJ*BK-AK*BJ
      YMUL=.5*XMUL/AREA
      QQ=AREA*XMUL*QGEN/4.
      QSTORE=AREA*XMUL*CV/4.
      MX(1)=K
      MX(2)=K+1
      IF (K.EQ.4) MX(2)=1
      MX(3)=5
      DO 155 I=1,3
      II=MX(I)
      QP(II)=QP(II)+QQ
      HC(II)=HC(II)+QSTORE
      DO 155 J=1,3
      JJ=MX(J)
  155 TC(II,JJ)=TC(II,JJ)+YMUL*E(I,J)
  150 CONTINUE
C
C     ***********************************************************************
C*****      ELIMINATE CENTER NODAL POINT
C     ***********************************************************************
C
      DO 160 I=1,4
      XMUL=TC(I,5)/TC(5,5)
      DO 160 J=1,4
  160 TC(I,J)=TC(I,J)-TC(J,5)*XMUL
      RETURN
      END
```

```
      SUBROUTINE SYMSOL (KKK)
C
C     ******************************************************************
C                SUBROUTINE TO INVERT BANDED SKEWED MATRICES
C     ******************************************************************
C
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/SYMARG/ NN,MM,A(500,32),B(500)
      GO TO (1000,2000), KKK
C
C     ******************************************************************
C*****     REDUCE MATRIX
C     ******************************************************************
C
 1000 DO 280 N=1,NN
      DO 260 L=2,MM
      C=A(N,L)/A(N,1)
      I=N+L-1
      IF (NN.LT.I) GO TO 260
      J=0
      DO 250 K=L,MM
      J=J+1
  250 A(I,J)=A(I,J)-C*A(N,K)
  260 A(N,L)=C
  280 CONTINUE
      GO TO 500
C
C     ******************************************************************
C*****     REDUCE VECTOR
C     ******************************************************************
C
 2000 DO 290 N=1,NN
      DO 285 L=2,MM
      I=N+L-1
      IF (NN.LT.I) GO TO 290
  285 B(I)=B(I)-A(N,L)*B(N)
  290 B(N)=B(N)/A(N,1)
C
C     ******************************************************************
C*****     BACK SUBSTITUTION
C     ******************************************************************
C
      N=NN
  300 N=N-1
      IF (N.EQ.0) GO TO 500
      DO 400 K=2,MM
      L=N+K-1
      IF (NN.LT.L) GO TO 400
```

```
      B(N)=B(N)-A(N,K)*B(L)
  400 CONTINUE
      GO TO 300
  500 RETURN
      END


      FUNCTION TFUNC (K,X,KK)
C
C     ********************************************************************
C*****      TEMPERATURE DEPENDENT THERMAL CONDUCTIVITY
C     ********************************************************************
C
      IMPLICIT REAL*8 (A-H,C-Z)
      TFUNC=1.
      RETURN
      END


      FUNCTION CFUNC (K,X)
C
C     ********************************************************************
C*****      TEMPERATURE DEPENDENT SPECIFIC HEAT AND DENSITY
C     ********************************************************************
C
      IMPLICIT REAL*8 (A-H,C-Z)
      CFUNC=1.
      RETURN
      END


      FUNCTION FCBC (TIME)
C
C     ********************************************************************
C*****      TIME DEPENDENT CONVECTION BOUNDARY CONDITIONS
C     ********************************************************************
C
      IMPLICIT REAL*8 (A-H,C-Z)
      FCBC=1.
      RETURN
      END
```

```
      FUNCTION FTBC (TIME)                                                385
C
C     ***************************************************************************
C*****     TIME DEPENDENT TEMPERATURE BOUNDARY CONDITIONS
C     ***************************************************************************
C
      IMPLICIT REAL*8 (A-H,C-Z)                                           386
      FTBC=1.                                                             387
      RETURN                                                              389
      END                                                                 390


      FUNCTION FFBC (TIME)                                                391
C
C     ***************************************************************************
C*****     TIME DEPENDENT FLUX BOUNDARY CONDITIONS
C     ***************************************************************************
C
      IMPLICIT REAL*8 (A-H,C-Z)                                           392
      FFBC=1.                                                             393
      RETURN                                                              394
      END                                                                 395
```

THESIS CHECK PROBLEM HEAT CONDUCTION IN A RECTANGLE

UNITS: FEET POUND MASS MINUTES FAHRENHEIT

TWO-DIMENSIONAL PLANE BODY
*************** ***** ****

NUMBER OF NODAL POINTS-- 36
NUMBER OF ELEMENTS------ 25
NUMBER OF CONVECTION BC- 0
NUMBER OF MATERIALS----- 1
NUMBER OF TIME SEQUENCES 1
OUTPUT PRINT INTERVAL--- 1
REFERENCE TEMPERATURE 1.000D 02 FAHRENHEIT

MATERIAL PROPERTIES
******** **********

MATERIAL	KXX	KYY	KXY	C	Q
1	8.66000D-01 BTU/MIN-FT-F	8.66000D-01 BTU/MIN-FT-F	0.0 BTU/MIN-FT-F	5.93000D 01	0.0

TIME SEQUENCING INFORMATION
**** ********** ***********

TIME SEQUENCE	TEMP. DEP. PROP.	TIME DEP.B.C.	NO. OF TIME STEPS	TIME INCREMENT
1	F	F	250	1.00000D-01 MINUTES

NODAL POINT INFORMATION
***** ***** ***********

N.P. NO.	CODE	X		Y		T	
1	1	0.0	FEET	0.0	FEET	1.0000000D 02	FAHRENHEIT
2	1	1.0000000D-01	FEET	0.0	FEET	1.0000000D 02	FAHRENHEIT
3	1	2.0000000D-01	FEET	0.0	FEET	1.0000000D 02	FAHRENHEIT
4	1	3.0000000D-01	FEET	0.0	FEET	1.0000000D 02	FAHRENHEIT
5	1	4.0000000D-01	FEET	0.0	FEET	1.0000000D 02	FAHRENHEIT
6	1	5.0000000D-01	FEET	0.0	FEET	1.0000000D 02	FAHRENHEIT
7	1	0.0	FEET	1.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
8	0	1.0000000D-01	FEET	1.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
9	0	2.0000000D-01	FEET	1.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
10	0	3.0000000D-01	FEET	1.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
11	0	4.0000000D-01	FEET	1.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
12	0	5.0000000D-01	FEET	1.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
13	1	0.0	FEET	2.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
14	0	1.0000000D-01	FEET	2.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
15	0	2.0000000D-01	FEET	2.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
16	0	3.0000000D-01	FEET	2.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
17	0	4.0000000D-01	FEET	2.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
18	0	5.0000000D-01	FEET	2.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
19	1	0.0	FEET	3.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
20	0	1.0000000D-01	FEET	3.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
21	0	2.0000000D-01	FEET	3.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
22	0	3.0000000D-01	FEET	3.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
23	0	4.0000000D-01	FEET	3.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
24	0	5.0000000D-01	FEET	3.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
25	1	0.0	FEET	4.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
26	0	1.0000000D-01	FEET	4.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
27	0	2.0000000D-01	FEET	4.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
28	0	3.0000000D-01	FEET	4.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
29	0	4.0000000D-01	FEET	4.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
30	0	5.0000000D-01	FEET	4.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
31	1	0.0	FEET	5.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
32	0	1.0000000D-01	FEET	5.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
33	0	2.0000000D-01	FEET	5.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
34	0	3.0000000D-01	FEET	5.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
35	0	4.0000000D-01	FEET	5.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT
36	0	5.0000000D-01	FEET	5.0000000D-01	FEET	1.0000000D 02	FAHRENHEIT

ELEMENT DESCRIPTION

ELEMENT	I	J	K	L	MATERIAL	HEAT GENERATION
1	7	8	2	1	1	0.0
2	8	9	3	2	1	0.0
3	9	10	4	3	1	0.0
4	10	11	5	4	1	0.0
5	11	12	6	5	1	0.0
6	13	14	8	7	1	0.0
7	14	15	9	8	1	0.0
8	15	16	10	9	1	0.0
9	16	17	11	10	1	0.0
10	17	18	12	11	1	0.0
11	19	20	14	13	1	0.0
12	20	21	15	14	1	0.0
13	21	22	16	15	1	0.0
14	22	23	17	16	1	0.0
15	23	24	18	17	1	0.0
16	25	26	20	19	1	0.0
17	26	27	21	20	1	0.0
18	27	28	22	21	1	0.0
19	28	29	23	22	1	0.0
20	29	30	24	23	1	0.0
21	31	32	26	25	1	0.0
22	32	33	27	26	1	0.0
23	33	34	28	27	1	0.0
24	34	35	29	28	1	0.0
25	35	36	30	29	1	0.0

TIME: 10.00000D-02 MINUTES

1	0.0	2	0.0	3	0.0	4	0.0	5	0.0	6	0.0
7	0.0	8	8.093000D 01	9	8.782146D 01	10	8.847629D 01	11	8.854166D 01	12	8.854769D 01
13	0.0	14	8.782146D 01	15	9.754114D 01	16	9.856728D 01	17	9.867569D 01	18	9.868602D 01
19	0.0	20	8.847629D 01	21	9.856728D 01	22	9.970710D 01	23	9.983385D 01	24	9.984634D 01
25	0.0	26	8.854166D 01	27	9.867569D 01	28	9.983385D 01	29	9.996554D 01	30	9.997879D 01
31	0.0	32	8.854769D 01	33	9.868602D 01	34	9.984634D 01	35	9.997879D 01	36	9.999219D 01

BIBLIOGRAPHY

1. Bliss, G. A., Calculus of Variations, The Mathematical Association of America, 1925.

2. Carslaw, H. S. and Jaeger, J. C., Conduction of Heat in Solids, 2nd ed., Oxford at the Clarendon Press, 1959.

3. Chapman, A. J., Heat Transfer, 2nd ed., The Macmillian Company, 1967.

4. Gurtin, M. E., "Variational Principles for Linear Initial Value Problems," Quarterly of Applied Mathematics, v22, p. 252-256, 1964.

5. IBM System/360, "FORTRAN IV Language," Form C28-6515-5, IBM Corporation, 1965.

6. Rhom and Haas Company Redstone Research Laboratories Report S-117, Application of the Finite Element Method to Heat Conduction in Solids, by E. B. Becker and C. H. Parr, November 1967.

7. Wilson, E. L. and Nickell, R. E., "Application of the Finite Element Method to Heat Conduction Analysis," Nuclear Engineering and Design, v4, p. 276-286, 1966.

8. Zienkiewicz, O. C. and Cheung, Y. K., "Finite Elements in the Solution of Field Problems," The Engineer, p. 507-510, September 24, 1965.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	20
2.	Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
3.	Professor Paul Pucci Department of Mechanical Engineering Naval Postgraduate School Monterey, California 93940	2
4.	Professor Gilles Cantin Department of Mechanical Engineering Naval Postgraduate School Monterey, California 93940	2
5.	LTJG Allan Bischof Chaloupka 1126 25th Street Greeley, Colorado	3
6.	Department of Mechanical Engineering Naval Postgraduate School Monterey, California 93940	2

Security Classification

DOCUMENT CONTROL DATA - R & D

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

1. ORIGINATING ACTIVITY (Corporate author)	2a. REPORT SECURITY CLASSIFICATION
Naval Postgraduate School Monterey, California 93940	Unclassified
	2b. GROUP

3. REPORT TITLE

A Computer Program for the Analysis of Two-Dimensional Heat Conduction Using the Finite Element Technique

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Master's Thesis

5. AUTHOR(S) (First name, middle initial, last name)

Allan Bischof Chaloupka

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
June 1969	85	8

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
b. PROJECT NO.	
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	

10. DISTRIBUTION STATEMENT

Distribution of this document is unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Naval Postgraduate School Monterey, California 93940

13. ABSTRACT

The computer program for the solution of unsteady heat conduction problems involving plane or axisymmetric geometry, devised by Robert E. Nickell, has been modified to include temperature dependent properties and time dependent boundary conditions. The original IBM 7094 computer dependent program has been converted for use on the IBM/OS 360 Model 67 computer. In both programs FORTRAN IV language was used. A "User's Manual" has been constructed for the modified program.

DD FORM 1 NOV 65 1473 (PAGE 1)

S/N 0101-807-6811

Security Classification

A-31408

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	W
Finite Element						
Heat Conduction						

DD FORM 1 NOV 65 1473 (BACK)
S/N 0101-807-6821