/2

TRANSIENT NONLINEAR HEAT TRANSFER USING FINITE ELEMENTS

J G KORVINK

B.Eng. (Mechanical Engineering)

RAU (Johannesburg)

September 1986

Submitted to the University of Cape Town
in partial fulfillment for the degree of
Master of Science in Engineering

I, Jan Gerrit Korvink submit this thesis in partial fulfillment of the requirements for the degree of Master of Science in Engineering. I claim that this is my original work and that it has not been submitted in this or similar form for a degree at any University.

SIGNED Signed by candidate

DATE ...2 October 1986....

ABSTRACT

This thesis is concerned with the numerical modelling of the transient nonlinear heat conduction problem in solid continua. The hyperbolic governing equation is specialised to a parabolic equation which is sufficient for most engineering applications. The theoretical development includes the effects of conduction, specific heat, internal heat generation and the boundary conditions of convection, radiation, specified temperatures and flux, as well as point sources in the domain. The finite element spatial semidiscretisation of the equations is formally derived from the weak form of the governing equations. Temporal discretisation is obtained through an implicit/explicit difference scheme. The material properties are allowed to be temperature dependent, and consequently a modified Newton-Raphson iterative scheme is employed to solve the equations. The fully discretised equations are solved by implementing the algorithm in an existing finite element stress analysis code. Modelling is possible using four or eight-noded isoparametric elements, and solution control is possible through choice of time step size and choice of time integration method. Five examples are employed to demonstrate the ability of the program. The results compare well with published analytical solutions.

ACKNOWLEDGEMENTS

I would like to express my gratitude to Professor J.B.Martin of the University of Cape Town for his guidance and constructive criticism.

Thanks is also expressed to my teachers and peers at the Applied Mechanics Research Unit, and in particular to Colin Mercer and Nick Marais, who in a cheerful manner gave more than their due.

Finally I wish to record my appreciation to the University of Cape Town and the Council for Scientific and Industrial Research for their financial support.

CONTENTS

Page

NOMENCLATURE

Roman capitals

- B - Gradient matrix
- $B(u,v)$ - Bilinear operator
- C_p - Specific heat at constant pressure
- C_v - Specific heat at constant volume
- J - Transformation Jacobian
- $J(u)$ - Energy functional
- K - Stiffness
- N - Shape function matrix
- P_i - Point source vector
- Q - Internal heat generation
- Q_g - Heat source
- R^n - Real space of n dimensions

Roman lower case

- $a_o,,a_n$ - Interpolation constants
- f - Finite element load vector
- g - Arbitrary function
- h - Enthalpy
- h_c - Convection coefficient

h_r - Radiation coefficient

k - Tensor of thermal conductivities

$k_o,,k_n$ - Interpolation constants

k_x - Principal thermal conductivity

k_y - Principal thermal conductivity

k_z - Principal thermal conductivity

$l(v)$ - Linear operator

p - Number of Gauss-Legendre sample points

p - Pressure

q - Heat flux

q_n - Neumann boundary flux

u - Internal energy

v - Specific volume

w_i - Gauss-Legendre weights

x - Problem domain coordinate

y - Problem domain coordinate

Greek capitals

Γ - Domain boundary

$\hat{\Gamma}$ - Master element boundary

Γ_d - Dirichlet boundary

Γ_n - Neumann boundary

Σ - Summation operator

θ - Temperature

$\dot{\theta}$ - Temperature rate of change

θ_c - Convection fluid temperature

θ_d - Dirichlet boundary temperature

θ_o - Initial temperature

θ_r - Radiation space temperature

θ^* - Arbitrary (dummy) temperature

Ω - Space in problem domain occupied by body

$\bar{\Omega}$ - Closure of Ω

$\hat{\Omega}$ - Master element domain

Greek lower case

α - Time integration parameter

ϵ - Surface emissivity

λ_i - Eigenvalues

ρ - Mass density

σ - Stephan-Boltzman constant

σ - Thermal relaxation time

τ - Time variable

ξ - Master element coordinates

η - Master element coordinates

μ - Specific heat per unit volume

κ - Radiation constant

Arrays

[C] - Finite element damping matrix

[I] - Identity matrix

[K] - Finite element stiffness matrix

$[K^*]$ - Modified finite element stiffness matrix

$[X]$ - Modal transformation matrix

$\{f\}$ - Finite element load vector

$\{f^*\}$ - Modified finite element load vector

$\{X\}$ - Eigenvector

$\{\Phi(\tau)\}$ - Modal state variable vector

$\{\lambda\}$ - Diagonal matrix of eigenvalues

$\{\theta\}$ - Nodal state variable vector (temperature)

$\{\theta_{n-1}\}$ - Ditto, for time step n-1

$\{\theta_n^i\}$ - Ditto, for time step n,iteration i

$\{\theta_n^{i+1}\}$ - Ditto, for time step n,iteration i+1

$\{\dot{\theta}\}$ - Nodal state variable rate vector (temperature rate)

$\{\dot{\theta}_{n-1}\}$ - Ditto, for time step n-1

$\{\dot{\theta}_n^i\}$ - Ditto, for time step n,iteration i

$\{\dot{\theta}_n^{i+1}\}$ - Ditto, for time step n,iteration i+1

Special symbols

Φ - Empty set

ϵ - Implies 'element of'

$\cap$ - Intersection operator

$\cup$ - Union operator

$\equiv$ - Implies 'equivalent to'

$\simeq$ - Approximately equal to (Truncated Taylor series sense)

$\pm$ - Argument following can take on both signs

R - Real line

R^3 - Real space

R^+ - Positive real line

Sym+ - Set of symmetric, positive definite matrices

$\forall$ - Implies 'for all'

∇ - Gradient operator

$[.]^T$ - Transpose, ie $[.]ij = [.]^T ji$

CHAPTER ONE: INTRODUCTION

1.1 Transient Thermal problems

The diffusion of heat in solids is an important engineering phenomenon, and the subject of a number of papers in international journals. The fields of application extend to most of the engineering disciplines (see for example J. Nuc. Eng. Des., Int. J. Heat and Mass Transfer, J. of Heat Transfer, Proc. ASME). The governing equations have been known for some time, but useful analytical solutions of the equations appeared much later [1] and are generally limited to a few simple cases. Often the solutions are in the form of Fourier series, with slow convergence rates and hence not very useful [2]. Analytical solutions become more scarce when the problem considered is nonlinear and transient. The introduction of convective and radiation boundary conditions, as well as the dependence of the material properties on the temperature are the causes of nonlinearity.

Clearly, to obtain reliable solutions for arbitrary problems (of arbitrary geometry and boundary conditions) the governing equations have to be discretised and solved numerically. Presently three major numerical solution techniques exist. The Finite Difference Method discretises and solves the governing partial differential equations. Discretisation of the weak form of the partial differential equations leads to the Finite Element Method. Finally, the Boundary Element Method is an approximation of the integral formulation of the partial differential equations. All three method are equivalent statements of the problem, and have successfully been applied to the steady-state linear conduction problem. This thesis uses the finite element method for spatial discretisation, and a difference scheme for temporal discretisation.

1.2 Finite element approximations

The finite element method has since its appearance been extensively applied to the field of stress analysis, particularly in structural analysis and continuum mechanics [3,4,5]. Having started as purely an engineer's tool, the collaboration of engineers and mathematicians has ensured the method of a sound theoretical basis [6].

The possibility of extending finite element analysis to solving general field problems came as early as 1964 [7], quickly leading researchers on to consider various potential field problems such as torsion and groundwater flow [8,9,10] as well as the diffusion of heat in a solid [11,12].

This is significant, as until the mid 1960's the predominant numerical method used in heat transfer analysis was the finite difference method. The advantages of the finite element method over the finite difference method are now well known, and include such features as adaptibility (to both constitutive and geometric nonlinearities) as well as numerical robustness , but most of all the ease with which finite elements may be used to solve transient and coupled problems [3].

1.3 Analogous problems

Of the possible first-order potential field problems encountered in nature, the most general one is that of heat conduction in a solid, due to the diversity of possible boundary conditions, and the existence of both a damping term as well as a source term.

PROBLEM	STATE VARIABLE	STIFFNESS	DAMPING	SOURCE	LOAD
HEAT CONDUCTION	TEMPERATURE	CONDUCTIVITY	SPECIFIC HEAT	INTERNAL HEAT GENERATION	FLUX
GAS DIFFUSION	CONCENTRATION	DIFFUSIVITY	1	-	MASS FLOW
ELECTRICAL CONDUCTION	VOLTAGE	CONDUCTIVITY	CAPACITANCE	VOLTAGE SOURCE	FLUX

Table 1.1: Equivalent First Order Transient Problems.

It is sensible then to establish a general solution method for heat conduction, and then to apply this method to the other potential field problems, as they are more likely to be subsets, or specialisations of the heat conduction problem.

The similarities between various potential field problems have been discussed by many authors [3,4]. It is possible to solve a variety of these problems using the present formulation once the equivalent terms of the governing equations have been identified. For some cases the equivalent terms can be found in table 1.1.

1.4 Form of the heat conduction equation.

It is interesting to note that the classical heat transport equation, which forms the basis of this thesis, contains a paradox in that the propagation speed of thermal signals is infinite. This is clearly unrealistic, and has caused a number of authors [13,14] to investigate and correct the equation. The result is a hyperbolic instead of a parabolic equation, and contains the second derivative of temperature with coefficient $\rho c \sigma$. This equation predicts a finite speed of heat propagation equal to $\sqrt{(k/\rho c \sigma)}$, (where k is the thermal conductivity, ρ the mass density and c the specific heat) provided $\sigma > 0$ (if $\sigma = 0$, we have an infinite speed as before). This phenomenon is referred to in the literature as 'second sound', as it attributes a wave-like motion to heat flow. The constant σ can be physically interpreted as the thermal relaxation time, ie the time lag needed to establish heat conduction in an element subjected to a sudden temperature change.

For metals, σ has been found to be of the order of 10^{-14}.

Clearly, since this constant is so small, the additional term $\frac{\partial^2\theta}{\partial \tau^2}$ can be neglected for most slow processes. There are, however, two practical cases where this term is important. When the flux is greater than 10^5 W/m², or when the elapsed time of a transient is less than 10^{-5} seconds, we should not neglect the 'acceleration' term. Examples where this occurs are nucleate boiling, laser penetration and welding, explosive bonding and welding. In problems involving very low temperatures (say around 1K), the full hyperbolic equation should also be used. Chandrasekharaiah [15] gives a detailed overview of this subject.

This thesis will only consider classical conduction of heat, and as a result will not be applicable to problems involving low absolute temperatures, short transients or high heat fluxes.

1.5 Discussion

The objectives of the thesis are, firstly, to establish the underlying principles of nonlinear, transient heat conduction in solids, which amounts to writing down the governing classical form of the law governing heat conduction in solids and categorising the (engineering) boundary conditions according to their mathematical equivalents. The second objective is to establish the weak form of the governing differential equation. The weak form limits the solution space of the problem, but remains mathematically equivalent. It does however lead naturally to the formulation of the Galerkin equations, and hence the finite element equations, which we need to solve general problems.

Having established the finite element equations, we next specialise the equations to isoparametric elements, because of the simplicity of mapping these elements onto any problem domain.

Following this, we discretise the equations in time, in order to solve transient problems. To maintain a certain degree of generality, we formulate the discretisation as a hybrid forward/backward difference scheme, controlled by a scalar parameter.

The final objective is to establish a framework within which these discretised problems can be solved. We wish to leave the framework as open and general as possible, to ensure that it can be expanded to accomodate future extensions and improvements easily. The numerical implementation is done using the program NOSTRUM [24], a general finite element package for the solution of NOnlinear STRuctural Mechanics problems. The work presented is an extension of the thesis of Pennington [16], who solved the steady-state conduction problem using an incremental formulation.

CHAPTER TWO: THEORY OF NONSTEADY-STATE HEAT TRANSFER

2.1 Preliminary theory (formal problem statement)

In this chapter we will state the transient thermal problem mathematically (after Wingett and Hughes [17]). From this statement, we will derive the classical form of the governing differential equation. The classical form will in turn lead us to the weak or variational form of the problem. With the initial value problem now in a suitable format, we will discretise the equations to yield the finite element equations. The final step is to adopt an isoparametric approach and to introduce the appropriate quadrature scheme, which will give us the equations to be solved numerically.

The Problem

We consider a body occupying Ω in R^3, where we understand R^3 to be real space. This implies that a point in Ω is defined by

$$x = \{x_i\} \quad ,i=1,2,3 \qquad [2.1]$$

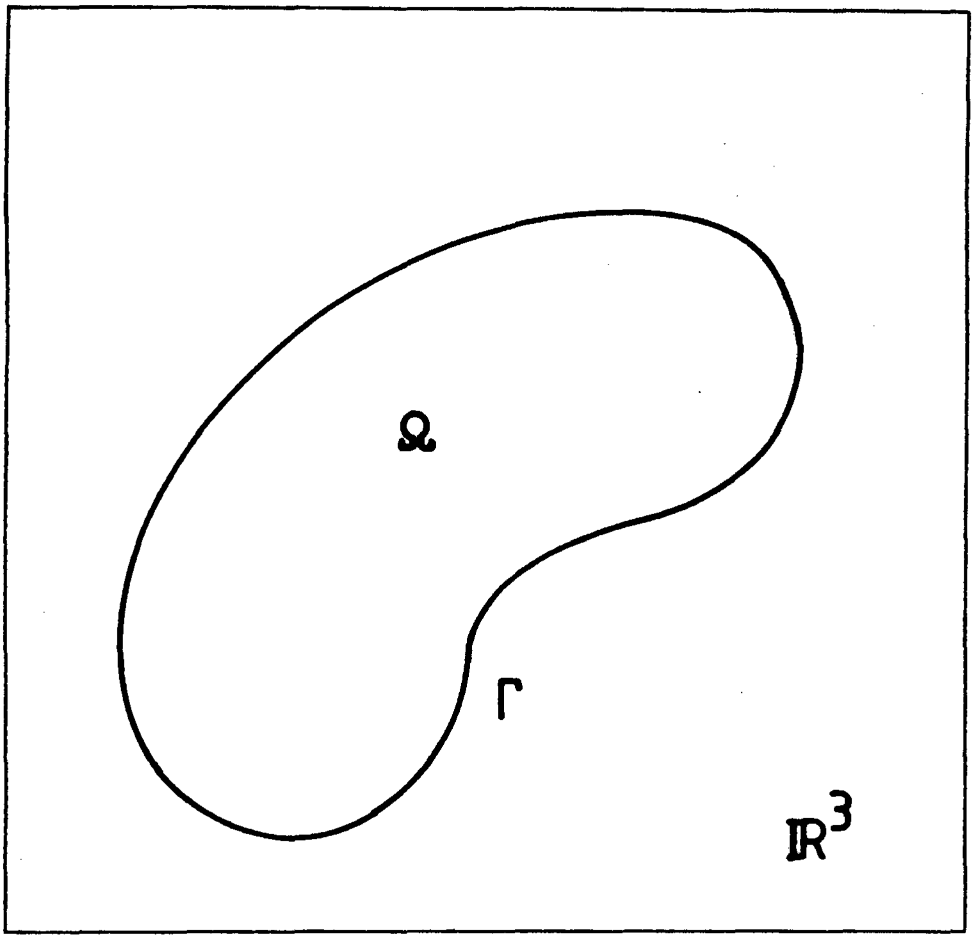


Figure 2.1: The problem domain

The region Ω is bounded by Γ, and Γ is assumed to be piecewise smooth. We will also allow Γ to be divided into two subdomains Γ_d, and Γ_n, with the restriction that

$$\Gamma = \overline{\Gamma_d \cup \Gamma_n}, \qquad [2.2a]$$

and

$$\Gamma_d \cap \Gamma_n = \phi, \qquad [2.2b]$$

ie, that Γ_d and Γ_n are disjoint.

The problem is understood to extend over the time period $\tau = 0$ to $\tau = \tau_f$, where $\tau \in R^+$.

The following properties will be specified for the material comprising the body:

$$\text{mass density} \qquad \rho: \quad \Omega \rightarrow R^+ \qquad [2.3a]$$

$$\text{specific heat} \qquad C_p: \quad R^+ \times \Omega \rightarrow R^+ \qquad [2.3b]$$

$$\text{thermal conductivity} \qquad k: \quad R^+ \times \Omega \rightarrow Sym^+ \qquad [2.3c]$$

Each of these properties may vary spatially within the body.

For most engineering solids the variation of density with temperature is usually of the second order, and hence ρ will not be a function of temperature; C_p and k may depend on temperature.

The last term that requires defining is the internal heat generation, which is, in the case of radioactive materials, a material property, but in most other cases, such as inductive heating, a result of the system being analysed. We define the quantity formally as:

$$\text{internal heat generation} \quad Q: \quad R^+ \times \Omega \times [0,\tau] \rightarrow R^+ \qquad [2.3d]$$

The Boundary conditions

On the boundaries we recognise the existence of both 'Dirichlet', or essential boundary conditions, as well as 'Neumann', or natural boundary conditions.

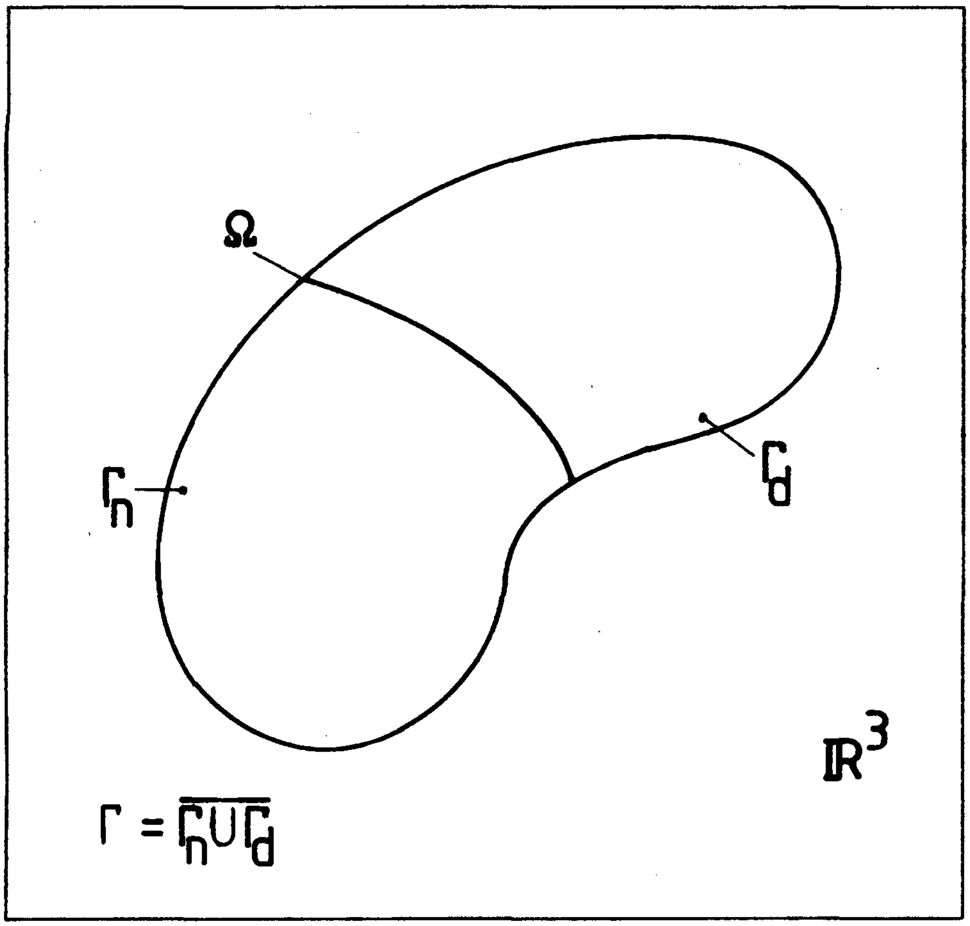


Figure 2.2: Subdivision of the boundary of the domain

The Dirichlet boundary conditions prescribe the temperature θ_d on Γ_d, which may vary both with position and time, ie.

$$\theta_d\colon \Gamma_d \times [0,\tau] \rightarrow R^+ \qquad [2.4a]$$

In the case of Neumann boundary conditions, a flux q_n is prescribed on Γ_n, which may also vary with position and time, ie

$$q_n : R^+ \times \Gamma_n \times [0,\tau] \to R^3 \qquad [2.4b]$$

The boundary conditions are complementary, which implies that either q_n or θ_d have to be specified.

Types of Neumann boundary conditions

The Neumann boundary can be expressed in a number of different forms, as shown in the figure below. These are now defined.

(i) Prescribed flux

The flux on some part of the boundary may be given a simple temporal and spatial dependence, ie

$$q_{n1} = q_t(x,\tau). \qquad [2.5a]$$

(ii) Adiabatic boundary

This is a special case of (i), where

$$q_{n2} = 0. \qquad [2.5b]$$

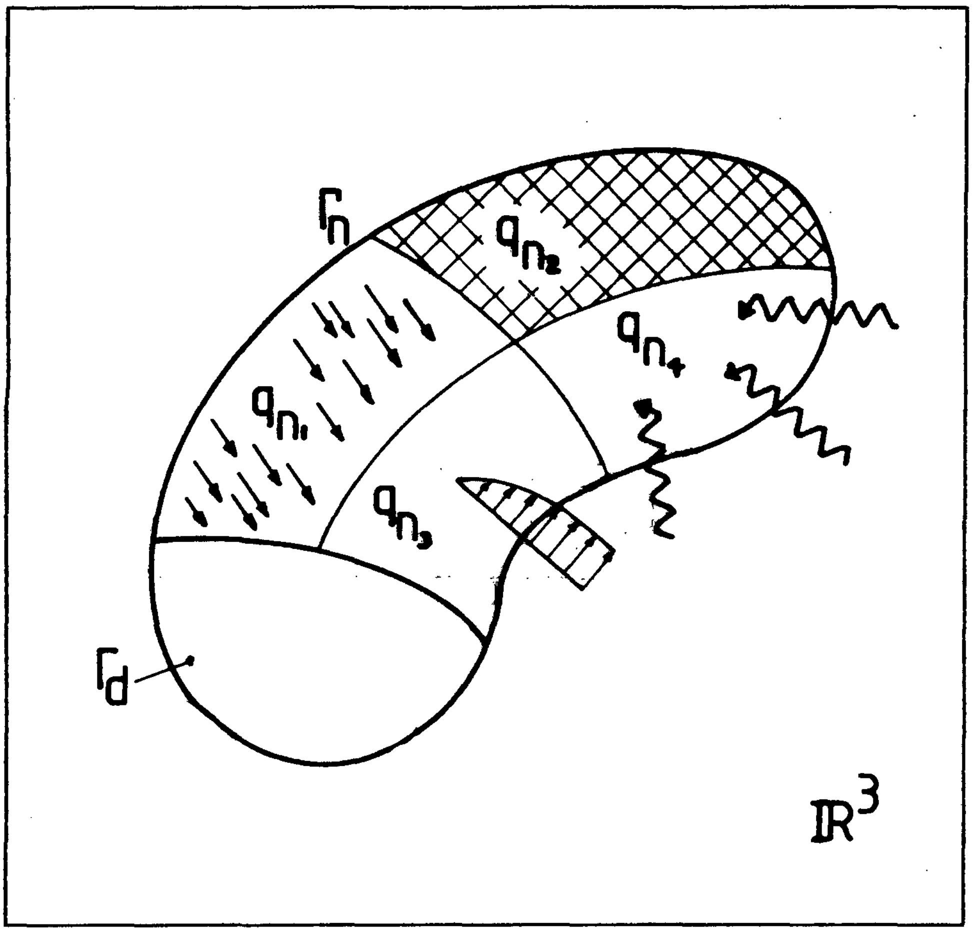


Figure 2.3: Admissible Neumann boundary conditions

(iii) Convection boundary condition

If the boundary of the domain Ω is in contact with some fluid, heat may cross the boundary Γ_n through the convection mechanism, ie

$$q_{n3} = h_c(x,\theta_c,\tau)(\theta(x,\tau) - \theta_c(x,\tau)). \qquad [2.5c]$$

Here h_c defines the convection coefficient of the interface, and

θ_c is the temperature field of the surrounding fluid. This boundary condition is sometimes referred to as a Cauchy-type boundary condition, especially when h_c is independent of temperature.

(iv) Radiation boundary conditions

Given the view factor of the boundary Γ_n with respect to the surrounding space, the emissivity of the boundary surface ϵ, as well as the Stephan-Boltzman constant σ, it is possible to determine a radiation coefficient h_r such that

$$q_{n4} = h_r(x,\tau)(\theta^4(x,\tau) - \theta_r^4(x,\tau)). \qquad [2.5d]$$

Here θ_r is the temperature field of the surrounding space.

The initial value

Since this is an initial value problem, we specify the temperature field of the body at time $\tau = 0$, ie

$$\theta_o : \Omega \rightarrow R^+. \qquad [2.6]$$

From the physical configuration it should be clear that this field need not be continuous.

2.2 Derivation of the Heat Conduction Equation.

To arrive at the classical form of the heat conduction equation, we will follow a purely mechanistic approach, that is, we will consider the physical situation directly, and perform an energy balance over an arbitrary element of the body. The derivation can be found in a number of texts [18,19,20] as the problem is of a fundamental nature. In order to keep the notation consistent, the derivation of Luikov ([18], p10) has been followed.

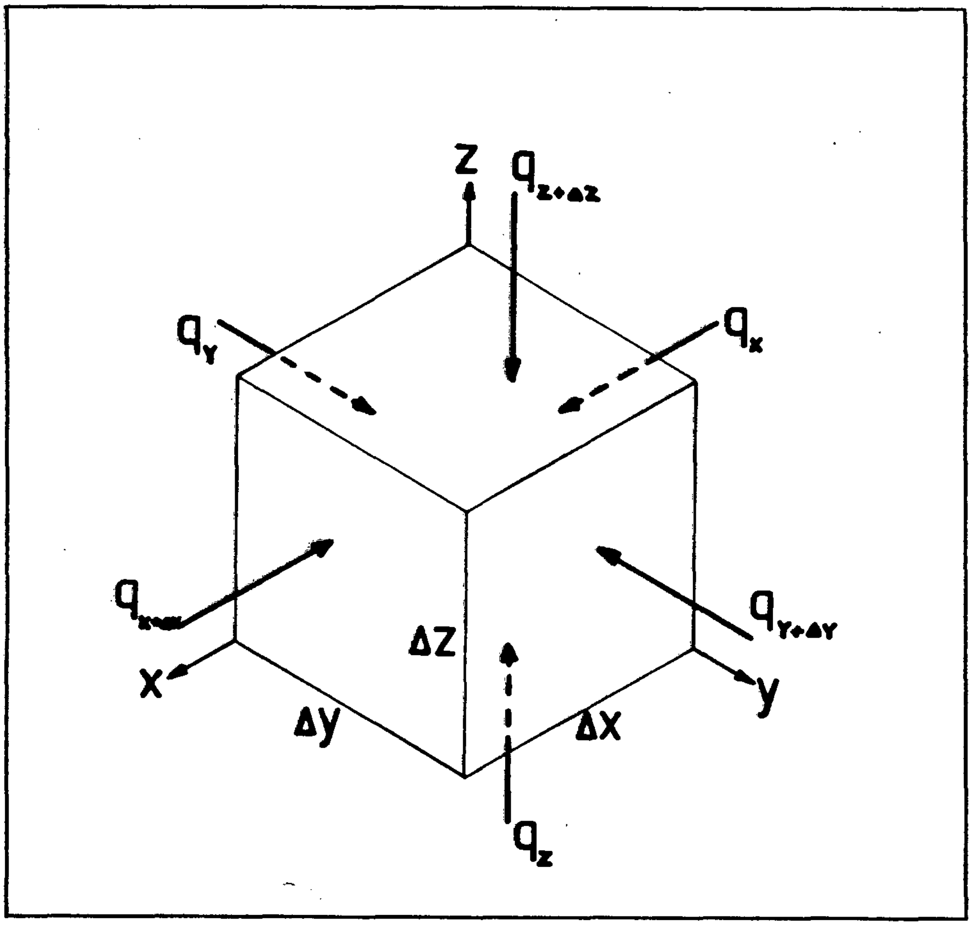


Figure 2.4: Heat flow balance on a material element

Consider a cubic differential material element as shown in figure 2.4.

The same properties have been considered as defined in the previous section:

ρ - density (effectively constant).

C_p - specific heat (assume we work within solid phase only).

k_x, k_y, k_z - principal heat conductivities in directions x, y and z.

Q^g - heat sources within element.

Now, summing the heat flows over the element, it follows that by the principle of energy conservation,

$$
\begin{aligned}
& Q^g + q_x \Delta y \Delta z - q_{x+\Delta x} \Delta y \Delta z \\
& + q_y \Delta z \Delta x - q_{y+\Delta y} \Delta z \Delta x \\
& + q_z \Delta x \Delta y - q_{z+\Delta z} \Delta x \Delta y \quad = C_p \rho \frac{\partial \Theta}{\partial \tau} \Delta x \Delta y \Delta z . \qquad [2.7]
\end{aligned}
$$

In the limit, where $\Delta x \rightarrow dx$, we have that

$$q_{x+\Delta x} \approx q_x + \frac{\partial q_x}{\partial x} dx,$$

$$q_{y+\Delta y} \approx q_y + \frac{\partial q_y}{\partial y} dy,$$

$$q_{z+\Delta z} \approx q_z + \frac{\partial q_z}{\partial z} dz. \qquad [2.8]$$

Substituting [2.8] into [2.7] results in:

$$Q^g - \frac{\partial q_x}{\partial x} dxdydz - \frac{\partial q_y}{\partial y} dxdydz - \frac{\partial q_z}{\partial z} dxdydz$$

$$= C_p \rho \frac{\partial \Theta}{\partial \tau} dxdydz. \qquad [2.9]$$

Introducing the constitutive relation [also known as Fourier's law),

$$q_i = -k_i \frac{\partial \Theta}{\partial x_i}, \quad (i = 1,2,3), \text{ (no sum on i)}, \qquad [2.10]$$

and rewriting the internal heat generation term in terms of the elemental volume,

$$Q^g = q^g dxdydz, \qquad [2.11]$$

we may rewrite [2.9] to yield the conduction equation

$$\frac{\partial}{\partial x}(k_x \frac{\partial \Theta}{\partial x}) + \frac{\partial}{\partial y}(k_y \frac{\partial \Theta}{\partial y}) + \frac{\partial}{\partial y}(k_y \frac{\partial \Theta}{\partial y}) = -q^g + C_p \rho \frac{\partial \Theta}{\partial \tau}. \qquad [2.12]$$

Finally the classical form can now be written as

$$\nabla.(k\nabla\theta) = -q^g + C_p\rho\frac{\partial\Theta}{\partial\tau} \text{ over } \Omega\times[0,\tau], \qquad [2.13]$$

with boundary conditions

$$\theta = \theta_d \qquad \text{on } \Gamma_d\times[0,\tau],$$

$$q = n.(k\nabla\theta) \quad \text{on } \Gamma_n\times[0,\tau],$$

$$\theta(x,0) = \theta_o \quad \forall\, x \in \Omega, \qquad [2.14]$$

where n is the unique outward normal associated with Γ_n, and all other symbols are as previously defined.

2.3 Formulation of heat transfer functional

We proceed along the normal route to obtain the weak form of [2.13]. This means choosing a test space S, defined as

$$S = \{\theta^*/\theta^* = \theta_d \text{ on } \Gamma_d\}. \qquad [2.15]$$

Let us form the functional Π by multiplying [2.13] by the arbitrary θ^*, and integrating over the domain. Note that whereas $\theta = \theta(x,\tau)$, $\theta^* = \theta^*(x)$, i.e. θ^* is not time dependent.

Thus:

$$\int_{\Omega} \{\nabla.(k\nabla\Theta)\}\Theta^{*}\, d\Omega = -\int_{\Omega} q^{g}\Theta^{*}\, d\Omega + \int_{\Omega} C_{p}\rho\dot{\Theta}\Theta^{*}\, d\Omega. \quad [2.16]$$

To get [2.15] into a more convenient form, we apply Green's theorem, i.e.

$$\int_{\Omega} \Phi\, \Psi_{,i}\, d\Omega = \oint_{\Gamma} \Phi\, \Psi\, \nu_{i}\, d\Gamma - \int_{\Omega} \Phi_{,i}\, \Psi\, d\Omega \qquad [2.17]$$

to the left hand side of equation [2.16], to get

$$\int \{\nabla.(k\nabla\Theta)\}\Theta^{*} d\Omega = \oint k\nabla\Theta.\Theta^{*} n d\Gamma - \int k\nabla\Theta.\nabla\Theta^{*} d\Omega \qquad [2.18]$$

From Fourier's law we have that

$$-k\nabla\Theta = q \qquad [2.19a]$$

and further we know that

$$q.n = q_{n} \qquad [2.19b]$$

Using [2.19a] and [2.19b] in [2.18], and replacing the left hand side of [2.16], we may write

$$\int_{\Omega} k\nabla\Theta.\nabla\Theta^{*} d\Omega + \int_{\Gamma} q_{n}\Theta^{*} d\Gamma = -\int_{\Omega} C_{p}\rho\dot{\Theta}\Theta^{*} d\Omega + \int_{\Omega} q^{g}\Theta^{*} d\Omega \quad [2.20]$$

Rewriting, we have equivalently that

$$\int_{\Omega} k\nabla\theta.\nabla\theta^{*}d\Omega + \int_{\Omega} C_{p}\rho\dot{\theta}\theta^{*}d\Omega = -\int_{\Gamma} q_{n}\theta^{*}d\Gamma + \int_{\Omega} q^{g}\theta^{*}d\Omega \quad [2.21]$$

We will use the above weak form to determine the finite-element equations. Dirichlet's principle (which states that the solution of $B(u,v) = l(u)$ is equivalent to minimising the energy functional $J(u) = \frac{1}{2}B(u,u) - l(u)$ often simplifies functional statements, but in the case of equation [2.21] leads to $\delta\dot{\theta}$ terms, which are undesirable.

2.4 Formulation of the Finite-Element Equations

To yield the final finite element equations, we first need to discretise the problem. This is accomplished by dividing the body Ω into n subbodies Ω_i (subdomains), and to ensure that the subdomains are mutually disjoint, we require that the following conditions be imposed:

$$\Omega_1^e \cup \Omega_2^e \cup \ldots \cup \Omega_i^e = \bar{\Omega},\ i = 1,,n, \quad [2.21a]$$

($\bar{\Omega}$ implies the closure of Ω, ie includes the boundary)

and

$$\Omega_1^e \cap \Omega_2^e \cap \ldots \cap \Omega_i^e = \Phi,\ i = 1,,n, \quad [2.21b]$$

We will assign m nodes to the discretised domain Ω. We further use a set of shape functions {N} as element basis functions, which will be used to interpolate nodal variables. These shape functions will have the usual property of being of value 1 at the associated node, and 0 at all other nodes.

We may now write the variables of [2.20] in terms of discrete nodal variables using the above shape functions:

$$\theta(x,\tau) = N_i\theta_i(\tau) \qquad [2.22a]$$

$$q(x,\tau) = N_i q_i(\tau) \qquad [2.22b]$$

It is easy to see that the following relations also hold,

$$\theta^*(x,\tau) = N_i\theta_i^*(\tau); \qquad [2.22c]$$

$$\dot\theta\ (x,\tau) = N_i\dot\theta_i(\tau); \qquad [2.22d]$$

$$\nabla\theta(x,\tau) = \nabla N_i\theta_i(\tau) = B_i\theta_i(\tau); \qquad [2.22c]$$

$$\nabla\theta^*(x,\tau) = \nabla N_i\theta_i^*(\tau) = B_i\theta_i^*(\tau). \qquad [2.22d]$$

Substituting [2.22] above into [2.20] yields

$$\int_\Omega kB_i\theta_i B_j\theta_j^* d\Omega + \int_\Omega C_p\rho N_i\dot\theta_i N_j\theta_j^* d\Omega$$

$$= \int_\Omega N_i q_i^g N_j\theta_j^* d\Omega - \int_\Gamma q_n N_j\theta_j^* d\Gamma. \qquad [2.23]$$

It should also be clear that [2.23] can be written equivalently as

$$\sum_{l=1}^{N} \int_{\Omega} kB_i^{(l)}\theta_i^{(l)}B_j^{(l)}\theta_j^{*(l)}d\Omega^{(l)}$$

$$+ \sum_{l=1}^{N} \int_{\Omega} C_p^{(l)}\rho^{(l)}N_i^{(l)}\dot{\theta}_i^{(l)}N_j^{(l)}\theta_j^{*(l)}d\Omega^{(l)}$$

$$= \sum_{l=1}^{N} \int_{\Omega} N_i^{(l)}q_i^{g(l)}N_j^{(l)}\theta_j^{*(l)}d\Omega^{(l)}$$

$$- \sum_{l=1}^{N} \int_{\Gamma} q_n^{(l)}N_j^{(l)}\theta_j^{*(l)}d\Gamma^{(l)}, \qquad [2.24]$$

where N is the number of elements. In matrix notation, for each element, we may now write

$$\theta^{*e}[\int_{\Omega e} B^{eT}k^eB^ed\Omega^e]\theta^e + \theta^{*e}[\int_{\Omega e} C_p^e\rho^eN^{eT}N^ed\Omega^e]\dot{\theta}^e$$

$$= \theta^{*e}[\int_{\Omega e} N^{eT}N^eq^gd\Omega^e] - \theta^{*e}[\int_{\Gamma e} N^{eT}q_nd\Gamma^e] \qquad [2.25].$$

Since θ^{*e} is arbitrary, and a common factor throughout, and θ^e and $\dot{\theta}^e$ are fixed nodal values that do not depend on x or y, we may rewrite [2.25] (dropping the e superscript)

$$\left[\int_{\Omega e} B^TkBd\Omega\right]\theta + \left[\int_{\Omega e} C_p\rho N^TNd\Omega\right]\dot{\theta}$$

$$= \left[\int_{\Omega e} N^TNq^gd\Omega\right] - \left[\int_{\Gamma e} N^Tq_nd\Gamma\right]. \qquad [2.26]$$

Let us consider now the last term on the right hand side of equation [2.26], as this is where we have to evaluate the Neumann boundary conditions.

As before we distinguish the same cases on the boundary, written in terms of the finite element interpolation matrices. This yields:

Specified Flux: $q_{n1} = q_t(\tau);$ [2.27a]

Convection: $q_{n2} = h_c(x,\tau)(\theta(x,\tau) - \theta_c(x,\tau))$

$= h_c(N.\theta - \theta_c(x,\tau));$ [2.27b]

and

Radiation: $q \quad = h_r(x,\tau)(\theta^4(x,\tau) - \theta_r^4(x,\tau))$

$= h_r(x,\tau)\theta^4(x,\tau) - \kappa q_{ri}$

$= h_r(N\theta N\theta N\theta N\theta) - \kappa N q_{ri}.$ [2.27c]

Next consider the first term on the right hand side of equation [2.26], i.e.

$$\int_{\Omega e} N^T N q^g d\Omega.$$

Let us here distinguish between internal heat generation,

$$q^g = G(\theta),\qquad [2.28]$$

and fixed point sources

$$P_e.\qquad [2.29]$$

As a result of [2.27] to [2.29], [2.26] becomes

$$\left[\int\!\!\int_{\Omega e} B^T kBd\Omega\right]\theta + \left[\int\!\!\int_{\Omega e} C_p \rho N^T Nd\Omega\right]\dot{\theta}$$

$$= \left[\int\!\!\int_{\Omega e} N^T NG(\theta)d\Omega + P_e\right]$$

$$- \left[\int\!\!\int_{\Gamma e} N^T h_c (N.\theta - \theta_c(x,\tau))d\Gamma\right.$$

$$+ \int_{\Gamma e} N^T (h_r (N\theta N\theta N\theta N\theta) - k_r Nq_{ri})d\Gamma$$

$$\left. + \int_{\Gamma e} N^T Nq_t d\Gamma\right] \qquad [2.30]$$

Let us rewrite equation [2.30], condensing integrals into symbols, i.e.

$$\{ K_c + K_h + K_r \}\theta + \{ C \}\dot{\theta}$$

$$= \{ f_g + f_{sp} + f_h + f_r + f_c + f_p \}, \qquad [2.31]$$

where

$$K_c = \int_{\Omega e} B^T kBd\Omega; \qquad \text{(conduction stiffness)}; \; [2.32a]$$

$$K_h = \int_{\Gamma e} N^T h_c Nd\Gamma; \qquad \text{(convection stiffness)}; \; [2.32b]$$

$$K_r = \int_{\Gamma e} N^T h_r N\theta N\theta N\theta N\theta d\Gamma; \text{(radiation stiffness)}; \; [2.32c]$$

$$f_g = \int_{\Omega e} N^T NG(\theta)d\Omega; \qquad \text{(generation vector)}; \; [2.32d]$$

$$f_p = P_e; \qquad \text{(point source vector)}; \; [2.32e]$$

$$f_c = \int_{\Gamma e} N^T h_c \theta_c(x,\tau)d\Gamma; \qquad \text{(convection vector); [2.32f]}$$

$$f_g = \int_{\Gamma e} N^T k_r N q_{ri} d\Gamma; \qquad \text{(radiation vector); [2.32h]}$$

$$f_{sp} = \int_{\Gamma e} N^T N q_t d\Gamma; \qquad \text{(specified flux vector); [2.32i]}$$

and finally,

$$C = \int_{\Omega e} C_p \rho N^T N d\Omega; \qquad \text{(Heat capacity matrix). [2.32j]}$$

Once assembled, we will refer to equation [2.21] in the following format,

$$\{ K \}\theta + \{ C \}\dot{\theta} = \{ f \} \qquad [2.33]$$

This is done both for simplicity, as well as the fact that the solution algorithms in the rest of the thesis use the information in this format.

2.5 Isoparametric Formulation

So far we have assumed the interpolation functions used within an element by referring to them merely as a vector N. We now choose these interpolation functions to be Lagrangian polynomials, for a number of reasons. Consider the following equation for the polynomials:

$$\hat{N}_i(\eta) = \frac{(\eta-\eta_1)\cdot\cdot(\eta-\eta_{i-1})(\eta-\eta_{i+1})\cdot\cdot(\eta-\eta_{k+1})}{(\eta_1-\eta_2)(\eta_1-\eta_3)\cdot\cdot(\eta_1-\eta_{k+1})} \qquad [2.34]$$

The polynomials defined above have the following property

$$\hat{N}_i(\eta_j) = \begin{bmatrix} 1 \text{ if } i=j \\ 0 \text{ if } i\neq j \end{bmatrix} \qquad [2.35]$$

This means that the (k+1) polynomials of equation [2.34] are linearly independent, which we require for the contributions of each of the nodal degrees of freedom. It further ensures that (in the Galerkin sense) the fixed constants are indeed the nodal variables, and hence the required nodal solution instead of just being arbitrary constants. A further advantage of these polynomials is that they can be chosen to be of any required order over the same domain.

To find the interpolation functions for a two-dimensional element, we simply multiply the polynomials as defined in the ξ-direction with those defined in the η-direction. Hence for four-noded elements we arrive at the following interpolation functions:

$$\hat{N}_i(\xi,\eta) = \frac{1}{4}(1+\xi\xi_i)(1+\eta\eta_i), \qquad [2.36a]$$

where we have for each node:

NODE	ξ_i	η_i
1	-1	-1
2	1	-1
3	1	1
4	-1	1

For eight-noded elements,

$$\hat{N}_i(\xi,\eta) = \frac{1}{4}(1+\xi\xi_i)(1+\eta\eta_i)(\xi\xi_i+\eta\eta_i-1),$$
$$i = 1,3,5,7;$$

$$\hat{N}_i(\xi,\eta) = \frac{1}{2}(1-\xi^2)(1+\eta\eta_i) \qquad i = 2,6$$
$$\hat{N}_i(\xi,\eta) = \frac{1}{2}(1-\eta^2)(1+\xi\xi_i) \qquad i = 4,8 \qquad [2.36b]$$

where for each node:

NODE	ξ_i	η_i
1	-1	-1
2	0	-1
3	1	-1
4	1	0
5	1	1
6	0	1
7	-1	1
8	-1	0

Next we wish to cover the problem domain with elements such that there will be no gaps left between the elements (this ensures continuity of the problem domain). If we use the same interpolation functions to determine the sides of the elements in the problem domain, we can enforce this requirement.

The means by which this is accomplished is the concept of an isoparametric element, which has a further advantage in that we are able to define all of the terms in the integrands of the various coefficient matrices and vectors on a regular master element. The only unknown contribution is the determinant of the mapping of this master element onto the problem domain. In planar problems this Jacobian is always a 2x2 matrix consisting of the derivatives of polynomials, and hence simple to evaluate.

The mapping of the master element onto the problem domain must be invertible, which amounts to ensuring that the Jacobian is positive definite, ie that the Jacobian determinant is greater than zero. This mapping is shown for two typical elements in figure 2.5 below. These elements are known as isoparametric elements as all nodal variables are interpolated by the same functions, including the nodal coordinates.

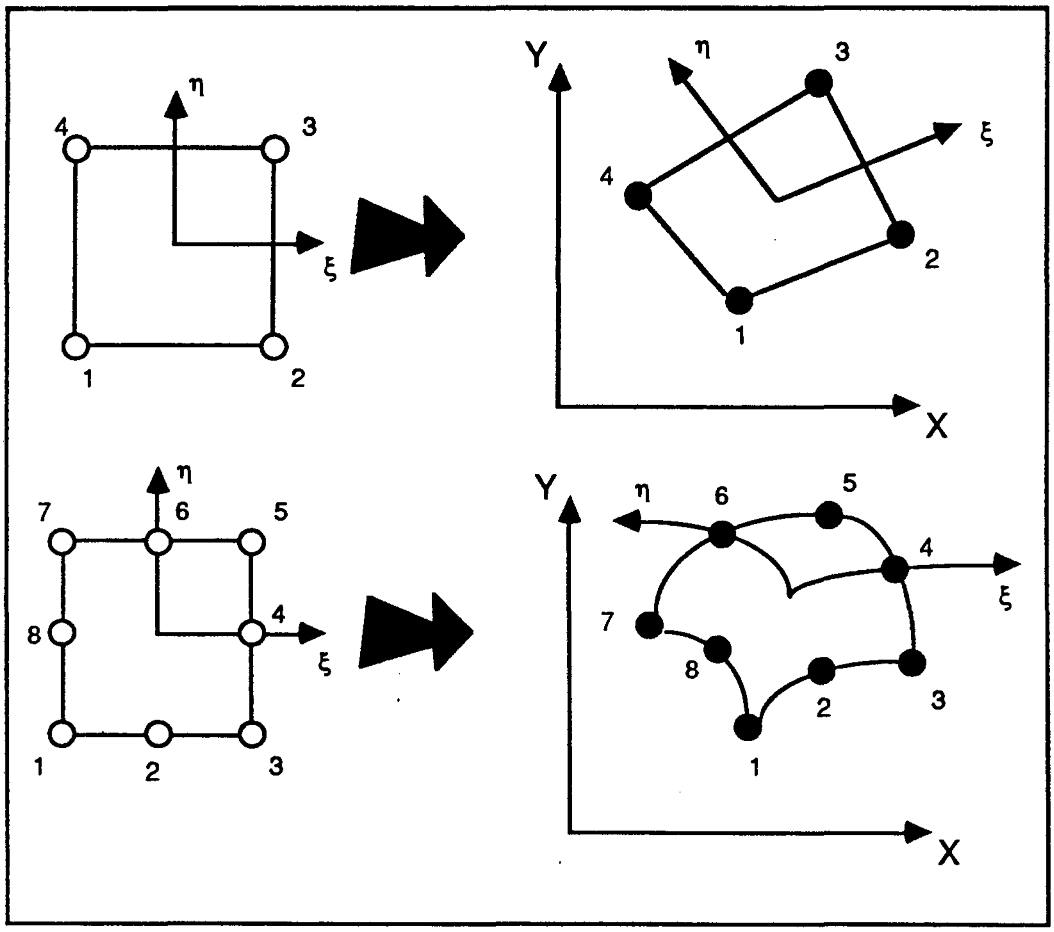


Figure 2.5: Mapping of master element onto problem domain

The transformation is characterised by the Jacobian determinant,

$$\det(J) = \frac{\partial(x,y)}{\partial(\xi,\eta)}, \qquad [2.37]$$

which is used to change variables in the integral equation, ie

$$d\Omega = |J|\, d\hat{\Omega} \qquad [2.38]$$

This allows us to rewrite equation [2.30] in terms of the master element coordinates (ξ,η),

$$\left[\int_{\hat{\Omega}} B^T k B\,|J|\,d\hat{\Omega}\right]\theta + \left[\int_{\hat{\Omega}} C_p \rho N^T N\,|J|\,d\hat{\Omega}\right]\dot{\theta}$$

$$= \left[\int_{\hat{\Omega}} N^T G(\theta)\,|J|\,d\hat{\Omega} + P_e\right]$$

$$- \left[\int_{\hat{\Gamma}} N^T h_c(N.\theta - \theta_c(x,\tau))\,|J|\,d\hat{\Gamma}\right.$$

$$+ \int_{\hat{\Gamma}} N^T (h_r(N\theta N\theta N\theta N\theta) - k_r N q_{ri})\,|J|\,d\hat{\Gamma}$$

$$\left. + \int_{\hat{\Gamma}} N^T N q_t\,|J|\,d\hat{\Gamma}\right]. \qquad [2.39]$$

2.6 Numerical Integration

At this stage all the element matrices and vectors are defined as integrals over either the boundary (line integrals) or the domain of the master element. The terms in the integrals are either polynomials or rational fractions, and hence explicit integration is only possible for some of the simplest cases.

To be in a position to exploit the speed and power of numerical machines, we need to evaluate the integrals by some other method. We now proceed according to the following algorithm:

Replace a typical integral

$$I = \int g(\xi)d\xi \qquad [2.40]$$

by the following sum of products;

$$I_p = \sum_{i=1}^{p}(w_i \cdot g(\xi_i)), \qquad [2.41]$$

where p is the number of evaluation points (which we may choose),

w_i is the weighting factor associated with

ξ_i, the position of the evaluation point.

The values of w_i and the positions of ξ_i are now selected to maximise the accuracy of the approximate integral. By adopting the master element approach, we have changed the limits of integration of all the integrals to -1 and +1, which simplifies the subsequent formulation.

The most efficient integration rules are those of Newton-Cotes and

Gauss-Legendre. It is worth noting that by choosing $w_1 = \frac{1}{3}$, $w_2 = \frac{4}{3}$, $w_3 = \frac{1}{3}$, and $\xi_1 = 1$, $\xi_2 = 0$, $\xi_3 = -1$, the integration scheme is exactly Simpson's rule.

The first six Gauss-Legendre rules are given in Table 2.1. The first three are by far the most widely used rules in finite element analysis.

Weight w_i	Position ξ_i
2.00000 00000 00000	0.00000 00000 00000
1.00000 00000 00000	$\pm$0.57735 02691 89626
0.55555 55555 55555 0.88888 88888 88888	$\pm$0.77459 66692 41483 0.00000 00000 00000
0.34785 48451 37454 0.65214 51548 62546	$\pm$0.86113 63115 94053 $\pm$0.33998 10435 84856
0.23692 68850 56189 0.47862 86704 99366 0.56888 88888 88889	$\pm$0.90617 98459 38664 $\pm$0.53846 93101 05683 0.00000 00000 00000
0.17132 44923 79170 0.36076 15730 48139 0.46791 39345 72691	$\pm$0.93246 95142 03152 $\pm$0.66120 93864 66265 $\pm$0.23861 91860 83179

Table 2.1: Commonly used Gauss-Legendre Quadrature rules

The extension of the one-dimensional formula to more dimensions is natural, and for the case of two dimensions amounts to:

$$\int\int g(\xi,\eta)d\xi d\eta = \int\left[\sum_{i=1}^{n} w_i\ g(\xi_i,\eta)\right]d\eta$$

$$= \sum_{i=1}^{n}\ \sum_{j=1}^{m} w_i\ w_j\ g(\xi_i,\eta_j). \qquad [2.42]$$

It is usual (but not necessary) to choose n = m. It is important to note that a p-point Gauss-Legendre rule will integrate polynomials up to order 2p-1 exactly. Hence a three point rule is sufficient for evaluating the matrices of a quadratic element, where the matrix entries are polynomials up to the fourth order in any one variable. For two-dimensional elements, we illustrate the integration points of the first three Gauss-Legendre rules in figure 2.6. The first rule is not an option in the present formulation.

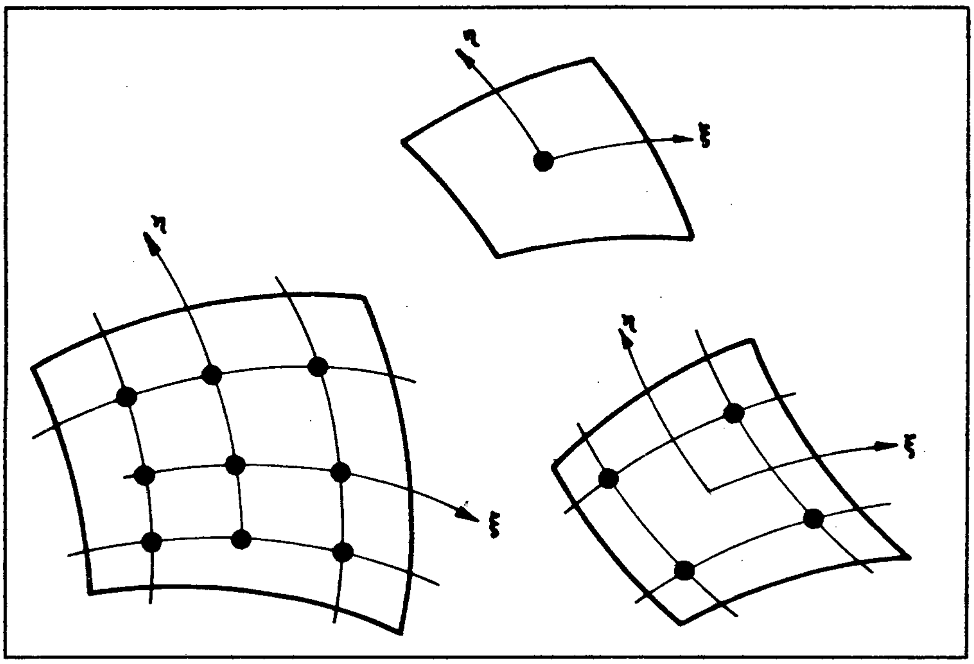


Figure 2.6: Gauss-Legendre integration points for 2-Dimensional elements

Replacing all the integrals in equation [2.39] by the quadrature formula results in:

$$\left[\sum_{i=1}^{n}\sum_{j=1}^{n} w_i\, w_j\, B^T(\xi_i,\eta_j)k(\xi_i,\eta_j)B(\xi_i,\eta_j)\,|J(\xi_i,\eta_j)|\right]\theta$$

$$+\left[\sum_{i=1}^{n}\sum_{j=1}^{n} w_i\, w_j\, C_p\rho N^T(\xi_i,\eta_j)N(\xi_i,\eta_j)\,|J(\xi_i,\eta_j)|\right]\dot{\theta}$$

$$= \left[\sum_{i=1}^{n}\sum_{j=1}^{n} w_i\, w_j\, N^T(\xi_i,\eta_j)G(\theta)\,|J(\xi_i,\eta_j)|\right.$$
$$+\sum_{e=1}^{p} P_e$$

$$-\sum_{k=1}^{sides}\sum_{i=1}^{n} w_i\, N^T(\xi_i,1)h_c\{N(\xi_i,1).\theta$$
$$-\theta_c(x,\tau)\}\,|J(\xi_i,1)|$$

$$+\sum_{k=1}^{sides}\sum_{i=1}^{n} w_i\, N^T(\xi_i,1)\ \{h_r(N\theta N\theta N\theta N\theta)$$
$$-k_r N(\xi_i,1)q_{ri}\}\,|J(\xi_i,1)|$$

$$\left.+\sum_{k=1}^{sides}\sum_{i=1}^{n} w_i\, N^T(\xi_i,1)N(\xi_i,1)q_t\,|J(\xi_i,1)|\right] \qquad [2.43]$$

Equation 2.43 can now be evaluated numerically on a digital computer. Note that equation [2.43] is of the form

$$[K]\{\theta\} + [C]\{\dot{\theta}\} = \{f\}, \qquad [2.44]$$

subject to the initial condition

$$\{\theta(0)\} = \{\theta_o\},$$

where $$[K] = \left[\sum_{i=1}^{n} \sum_{j=1}^{n} w_i\ w_j\ B^T(\xi_i,\eta_j)k(\xi_i,\eta_j)B(\xi_i,\eta_j)\,|J(\xi_i,\eta_j)|\right]$$

$$+ \sum_{k=1}^{sides} \sum_{i=1}^{n} w_i\ N^T(\xi_i,1)h_c\{N(\xi_i,1)$$

$$+ \sum_{k=1}^{sides} \sum_{i=1}^{n} w_i\ N^T(\xi_i,1)\ \{h_r(N\theta N\theta N\theta N),$$

$$[C] = \sum_{i=1}^{n} \sum_{j=1}^{n} w_i\ w_j\ C_p\rho N^T(\xi_i,\eta_j)N(\xi_i,\eta_j)\,|J(\xi_i,\eta_j)|,$$

and $$\{f\} = \left[\sum_{i=1}^{n} \sum_{j=1}^{n} w_i\ w_j\ N^T(\xi_i,\eta_j)G(\theta)\,|J(\xi_i,\eta_j)|\right.$$

$$+ \sum_{e=1}^{p} P_e$$

$$+ \sum_{k=1}^{sides} \sum_{i=1}^{n} w_i\ N^T(\xi_i,1)h_c\{\theta_c(x,\tau)\}\,|J(\xi_i,1)|$$

$$- \sum_{k=1}^{sides} \sum_{i=1}^{n} w_i\ N^T(\xi_i,1)\ \{k_r N(\xi_i,1)q_{ri}\}\,|J(\xi_i,1)|$$

$$\left.+ \sum_{k=1}^{sides} \sum_{i=1}^{n} w_i\ N^T(\xi_i,1)N(\xi_i,1)q_t\,|J(\xi_i,1)|\right].$$

CHAPTER THREE: DISCRETISATION IN TIME AND ITERATIVE PROCEDURES

3.1 Formulation of time-difference scheme

Applying the finite element method to the heat conduction equation has led to a semi-discrete matrix differential equation of the first order,

$$[K]\{\theta\} + [C]\{\dot{\theta}\} = \{f\}, \ \forall \ \tau > 0,$$

$$\{\theta(\tau_o)\} = \{\theta_o\}, \qquad [3.1]$$

where [K] is the symmetric, semi-positive definite stiffness or conductivity matrix;
[C] is the symmetric, positive definite damping or thermal capacitance matrix;
{f} is the thermal load vector;
{θ} is the vector of nodal (temperature) unknowns;
and {$\dot{\theta}$} is the first time derivative of {θ}.

Equation 3.1 represents a coupled system of m ordinary first order differential equations, where m is equal to the number of non-constrained nodes.

For a linear system, [K], [C] and {f} are independent of $\{\Theta\}$ and $\{\dot{\Theta}\}$. In most physical problems, [K] and [C] are independent of the time. In nonlinear systems, [K] and [C] are dependent on $\{\Theta\}$.

A rich variety of methods exists to solve equation 3.1, but these may be broadly classified as follows:

a. Mode superposition methods (linear problems only),

b. Direct integration methods.

In modal superposition methods the approach is firstly to decouple equation 3.1 into modal equations, using the transformation

$$\{\Theta(\tau)\} = [X]\{\Phi(\tau)\}, \qquad [3.2]$$

where the transformation matrix [X] is made up of the m eigenvectors as defined by

$$([K] - \lambda_i[C])\{X_i\} = 0 \qquad i=1,2,..,m$$

$$[X] = [\{X_1\}\{X_2\}...\{X_m\}]. \qquad [3.3]$$

Using 3.2, equation 3.1 becomes

$$[X]^T[K][X]\{\Phi(\tau)\} + [X]^T[C][X]\{\dot{\Phi}(\tau)\} = [X]^T\{f(\tau)\}. \qquad [3.4]$$

The matrix [X] satisfies the following orthogonality requirements:

$$[X]^T[C][X] = [I]$$
$$[X]^T[K][X] = [\lambda]. \qquad [3.5]$$

Substituting 3.5 into 3.4 leads to

$$\{\dot{\phi}(\tau)\} + [\lambda]\{\phi(\tau)\} = \{\overline{F}(\tau)\}, \quad \forall\ \tau > \tau_o. \qquad [3.6]$$

Note that equation 3.6 is fully decoupled, which allows each equation to be integrated independently. To yield the final results, the modes are superimposed using equation 3.2.

Direct integration methods lead equation 3.1 to become fully discretised, and generally result in a modified equation of the form

$$[K^*]\{\Theta\} = \{f^*\}, \qquad [3.7]$$

where $[K^*]$ is an effective stiffness matrix,
$\{f^*\}$ is an effective load vector, and
$\{\Theta\}$ is the vector of nodal unknowns.

The modified equation may now be solved using standard techniques such as the skyline solver or the frontal method devised by Irons.

The direct integration methods may be further subdivided into explicit and implicit methods, as well as single and multistep families.

Multistep methods are popular because the prediction of a future value depends on the results of a number of previous time steps weighted together according to a precise formula. This results in very accurate and stable predictions. These methods often have additional predictor/corrector formula pairs which improve accuracy even further.

Multistep methods have two disadvantages, one being that they require a number of time solutions to get started (ie they are not self starting), the other being that a number of time histories of the variable have to be stored at each time step for the prediction at the next time interval. In finite element applications we deal with a vector of variables at each time step, and storing these vectors becomes prohibitive.

Single step methods avoid these difficulties, which is probably the reason that they are so popular with finite element researchers.

It is possible to combine the forward, central and backward difference schemes into one formulation, better known as the semi-implicit Euler's method, or the generalised trapezoidal rule. For this method we use a scalar multiplier α, to get:

$$\{\dot{\theta}_{\tau+\alpha\Delta\tau}\} = \alpha\{\dot{\theta}_{\tau+\Delta\tau}\} + (1-\alpha)\{\dot{\theta}_{\tau}\}, \quad 0<\alpha<1. \qquad [3.8]$$

The choice of α leads to a number of well known and documented methods, as shown in table 3.1.

family	alpha	Method	Alternate name
explicit	0	forward difference	Forward Euler method
	1/2	central difference	Midpoint rule, Trapeziodal rule, Crank-Nicholson.
implicit	2/3	Galerkin	
	1	Backward difference	Backward Euler method

Table 3.1: Generalised Trapezoidal family of Difference Schemes

The explicit formulation leads to a 'lumped' diagonal effective stiffness, and needs no equation solving to advance in time (Hughes [21], p72). Equation 3.9 can be used in a number of ways to lead to very different $[K^*]$ and $\{f^*\}$ coefficient matrices.

3.2 General Solution Algorithms.

There are two major approaches to solving equation 3.1 (modified to equation 3.7) when using direct integration methods. The first is the incremental approach, and the second the total state variable approach, ie

$$[K^*]\{\theta_{T+\Delta T}\} = \{f^*_{T+\Delta T}\}. \qquad [3.9]$$

If we use equation 3.8 to approximate $\{\Theta\}$, it can be shown (Dhatt and Fouzott [22], p314) that for the coefficient matrices we have:

$$[K^*] = [C] + \alpha\Delta\tau[K]$$

$$\{f^*\} = \Delta\tau[\alpha\{f_{\tau+\Delta\tau}\} + (1-\alpha)\{f_\tau\} - (1-\alpha)[K]\{\Theta_\tau\}] + [C]\{\Theta_\tau\} \qquad [3.10]$$

Note that this formulation requires the storage of the load vector from the previous time step.

3.3 Nonlinear Algorithms

The solution of the equations becomes more difficult when the coefficient matrices are nonlinear, as we have to iterate at each time step to get to a solution.

A popular approach is the full Newton-Raphson scheme.

For very small increments in $\{\Theta\}$ we may write the truncated Taylor series

$$[K(\Theta_n^{i+1})] = [K(\Theta_n^i)] + \{\Delta\Theta_n^i\}^T \partial[K]/\partial\Theta \Big|_{\{\Theta_n^i\}}, \qquad [3.11]$$

and

$$[C(\Theta_n^{i+1})] = [C(\Theta_n^i)] + \{\Delta\Theta_n^i\}^T \partial[C]/\partial\Theta \Big|_{\{\Theta_n^i\}}, \qquad [3.12]$$

where $\{\theta_n^{i+1}\}$ is the temperature at iteration i+1, at time n;

$\{\theta_n^i\}$ is the temperature at iteration i, at time n;

$\{\Delta\theta_n^i\} = \{\theta_n^{i+1}\} - \{\theta_n^i\}$.

Using equations [3.11] and [3.12] it is now possible to rewrite equation [3.1] for the $(i+1)^{th}$ iteration

$$\left[[K(\theta_n^i)] + \{\Delta\theta_n^i\}^T \frac{\partial[K]}{\partial\theta}\right]\{\theta_n^{i+1}\}$$

$$+ \left[[C(\theta_n^i)] + \{\Delta\theta_n^i\}^T \frac{\partial[C]}{\partial\theta}\right]\{\dot{\theta}_n^{i+1}\} = \{f_n\} \qquad [3.13]$$

Finally, introducing the time integration-schemes

$$\theta_n^{i+1} = \theta_{n-1} + \Delta\tau\{(1-\alpha)\dot{\theta}_{n-1} + \alpha\dot{\theta}_n^{i+1}\}, \text{ and}$$

$$\theta_n^i = \theta_{n-1} + \Delta\tau\{(1-\alpha)\dot{\theta}_{n-1} + \alpha\dot{\theta}_n^i\}, \qquad [3.14]$$

and noting that

$$\begin{aligned}\Delta\theta_n^i &= \theta_n^{i+1} - \theta_n^i \\ &= \alpha\Delta\tau(\dot{\theta}_n^{i+1} - \dot{\theta}_n^i),\end{aligned} \qquad [3.15]$$

equation [3.13] becomes

$$\{\theta_n^{i+1}\}^T \left[\frac{\partial[K]}{\partial\theta}\bigg|_{\theta_n^i} + \frac{1}{\alpha\Delta\tau}\frac{\partial[C]}{\partial\theta}\bigg|_{\theta_n^i}\right]\{\theta_n^{i+1}\}$$

$$+ \left[K[\theta_n^i] - \theta_n^i \frac{\partial[K]}{\partial\theta}\bigg|_{\theta_n^i} + \frac{1}{\alpha\Delta\tau}[C(\theta_n^i)]\right.$$

$$\left. - \theta_n^i \frac{\partial[C]}{\partial\theta}\bigg|_{\theta_n^i} + \frac{1}{\alpha\Delta\tau}\theta_n^i \frac{\partial[C]}{\partial\theta}\bigg|_{\theta_n^i}\theta_{n-1}\right]\{\theta_n^{i+1}\}$$

$$= \left[\{f_n\} + \frac{1}{\alpha\Delta\tau}[C(\theta_n^i)] - \frac{1}{\alpha\Delta\tau}\theta_n^i \frac{\partial[C]}{\partial\theta}\bigg|_{\theta_n^i}\theta_{n-1}\right]. \quad [3.16]$$

We simplify equation [3.16] by writing

$$\{\theta_n^{i+1}\}^T[K_1^*]\{\theta_n^{i+1}\} + [K_2^*]\{\theta_n^{i+1}\} = \{f_n^*\}. \quad [3.17]$$

There are four major disadvantages to using the full Newton-Raphson scheme. The first is that we have a second order equation in terms of the state variable. The reformation of the stiffness matrix is complex, and requires the storage of a lot of information, which in finite element analysis becomes prohibitive. Next, we need to evaluate $\frac{\partial[K]}{\partial\theta}$ and $\frac{\partial[C]}{\partial\theta}$, and these terms could be complicated depending on the degree of nonlinearity present. Finally, the problem becomes even more complicated if $f_n = f(\theta_n)$, which is severly restrictive on the type of problems we may wish to attempt.

A more realistic scheme is the direct iterative approach, which is based in part on the Newton-Raphson method. For this scheme it is possible to follow either an incremental formulation (Pennington [16]), or a full state variable approach as we have done. This method eliminates the necessity for storing the stiffnesses of previous time steps. The simplification is possible if the following formula is adopted to approximate the temperature at a future time step:

$$\theta_n = \theta_{n-1} + \Delta\tau\{(1-\alpha)\dot{\theta}_{n-1} + \alpha\dot{\theta}_n\} \qquad [3.18]$$

Equation [3.1] is now rewritten to incorporate an iterative scheme, where the superscript i refers to the iteration number, and the subscript n refers to the time step:

$$[K(\theta_n^i)]\{\theta_n^{i+1}\} + [C(\theta_n^i)]\{\dot{\theta}_n^{i+1}\} = \{f(\theta_n^i)\}. \qquad [3.19]$$

Here $\theta_n^{i+1} \rightarrow \theta_n$ as i becomes large. To eliminate the $\dot{\theta}$ term, we rewrite equation [3.18], again introducing the iterative scheme:

$$\dot{\theta}_n^{i+1} = \frac{1}{\alpha\Delta\tau_n}(\theta_n^{i+1} - \theta_{n-1}) - \frac{(1-\alpha)}{\alpha}\dot{\theta}_{n-1}. \qquad [3.20]$$

Next we substitute equation [3.20] into equation [3.19] to yield

$$\left[[K(\theta_n^i)] + \frac{1}{\alpha\Delta\tau_n}[C(\theta_n^i)]\right]\{\theta_n^{i+1}\} =$$

$$\left[\{f(\theta_n^i)\} + [C(\theta_n^i)]\{\frac{1}{\alpha\Delta\tau_n}\{\theta_{n-1}\} + \frac{(1-\alpha)}{\alpha}\{\dot{\theta}_{n-1}\}\}\right], \qquad [3.21]$$

which may be condensed to

$$[K^*]\{\theta_n^{i+1}\} = \{f^*\}. \qquad [3.22]$$

We may now iterate using this equation until we have reached convergence, with the convergence requirement defined as

$$|\theta_n^{i+1} - \theta_n^{i}| < \text{tolerance}. \qquad [3.23]$$

3.4 Stability requirements

The aim of a numerical algorithm which integrates the finite element system equations is to evaluate a good approximation to the system under consideration. One requires the elimination of spurious oscillations which indicate numerical instabilities, as well as precision of the solution. As a rough estimate the time step length should be an order of magnitude smaller than the smallest period of the system [22].

The stability of an integration method is ascertained by examining the behaviour of the algorithm for arbitrary initial conditions. Considering the stability of integration methods, we have procedures that are unconditionally stable and that are only conditionally stable. An integration method is unconditionally stable if the solution for any initial conditions does not grow without bound for any time step $\Delta\tau$, particularly when $\frac{\Delta\tau}{T}$ is large. The method is only conditionally stable if the above only holds provided that $\frac{\Delta\tau}{T}$ is smaller than some stability limit.

The stability of the α-integration scheme is investigated by using equation 3.1 for a one degree of freedom system, ie

$$C_p\dot{\theta} + k\theta = 0, \qquad [3.24]$$

and solving for θ at time $\tau = n$ using equation 3.18, which yields

$$\theta_n = \frac{1 - (1 - \alpha)\Delta\tau(k_n)}{1 + \alpha\Delta\tau(k_n)}\,\theta_{n-1}. \qquad [3.25]$$

For stability, we require that for any time step

$$\left|\frac{1 - (1 - \alpha)\Delta\tau(k_n)}{1 + \alpha\Delta\tau(k_n)}\right| \leq 1, \qquad [3.26]$$

so that we limit the 'growth' of θ. This condition is satisfied for any $\Delta\tau$ provided that $\alpha \geq 0.5$. For $\alpha < 0.5$, we limit the time step length by

$$\Delta\tau = \frac{2}{(1 - 2\alpha)(k_n)}, \qquad [3.27]$$

for stability. This implies that the method is unconditionally stable for $\alpha \geq 0.5$ and conditionally stable for $\alpha < 0.5$. For reasonable accuracy, however, we may need to impose a further restriction on the time step length for both cases of α.

For multiple degrees of freedom systems we replace the value of k_n with the largest eigenvalue of the system [21]

$$([K] - \lambda[C])\phi = 0. \qquad [3.28]$$

The stability limit has been calculated for a typical rectangular element and can be found in Appendix A2.

CHAPTER FOUR: APPROXIMATING MATERIAL PROPERTIES

4.1 Description of heat transfer properties

The main source of nonlinearity in dealing with heat diffusion problems is the variation of the heat transfer properties with temperature. Other sources of nonlinearity exist as well, such as the dependence of the properties on the stress state as well as the deformation history (which can lead to local changes in properties). Furthermore, we find in the case of radiation boundary conditions, that the body has associated with each possible orientation a view factor which determines the maximum effective heat transfer area. Added to this, the amount of heat transferred depends on the fourth power of the temperature.

We will only consider the variation of the properties with temperature, and will assume other possible variations constant for the duration of the problem.

DENSITY

Density is defined as the mass per unit volume in the limit where the volume vanishes. In this thesis we adopt a continuum approach, which implies that the systems we consider are of such a size for statistical averages to be meaningful.

For most engineering materials, density is effectively constant with temperature while the materials are in the solid phase. As a result, the density has been left as a constant in the present formulation. Figure 4.1 shows the variation of density with temperature for a number of commonly used metals. Clearly the variation is less than 1% over the entire useful range.

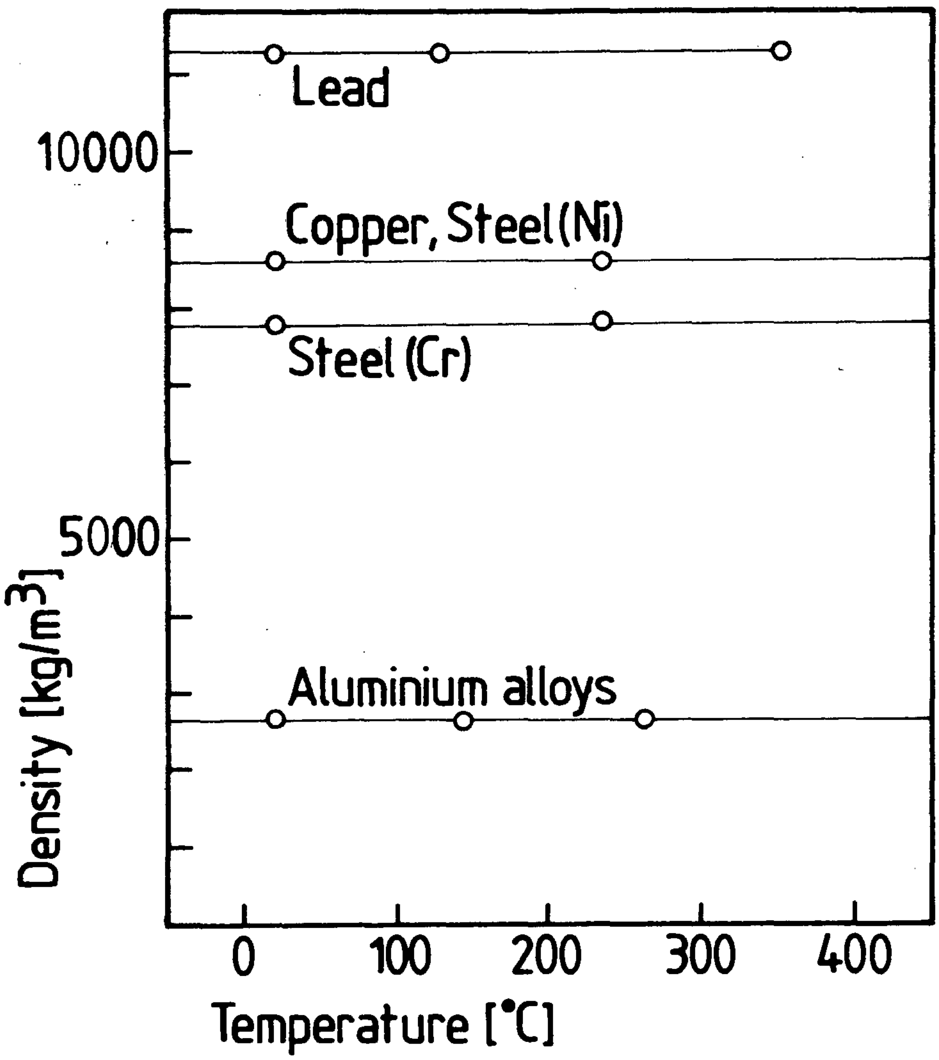


Figure 4.1: Densities of a number of common metals

For some materials, such as composite building materials with large flexible air-filled spaces and certain plastics, a more noticeable variation exists. Since we do not allow variations in density it is possible to circumvent this by making use of another property, the heat capacity per unit volume, defined by Biot as:

$$\mu = C_p \rho. \qquad [4.1]$$

The variation of the two terms is treated as one, and can be implemented in the present formulation by setting ρ equal to unity and replacing C_p with the function for μ.

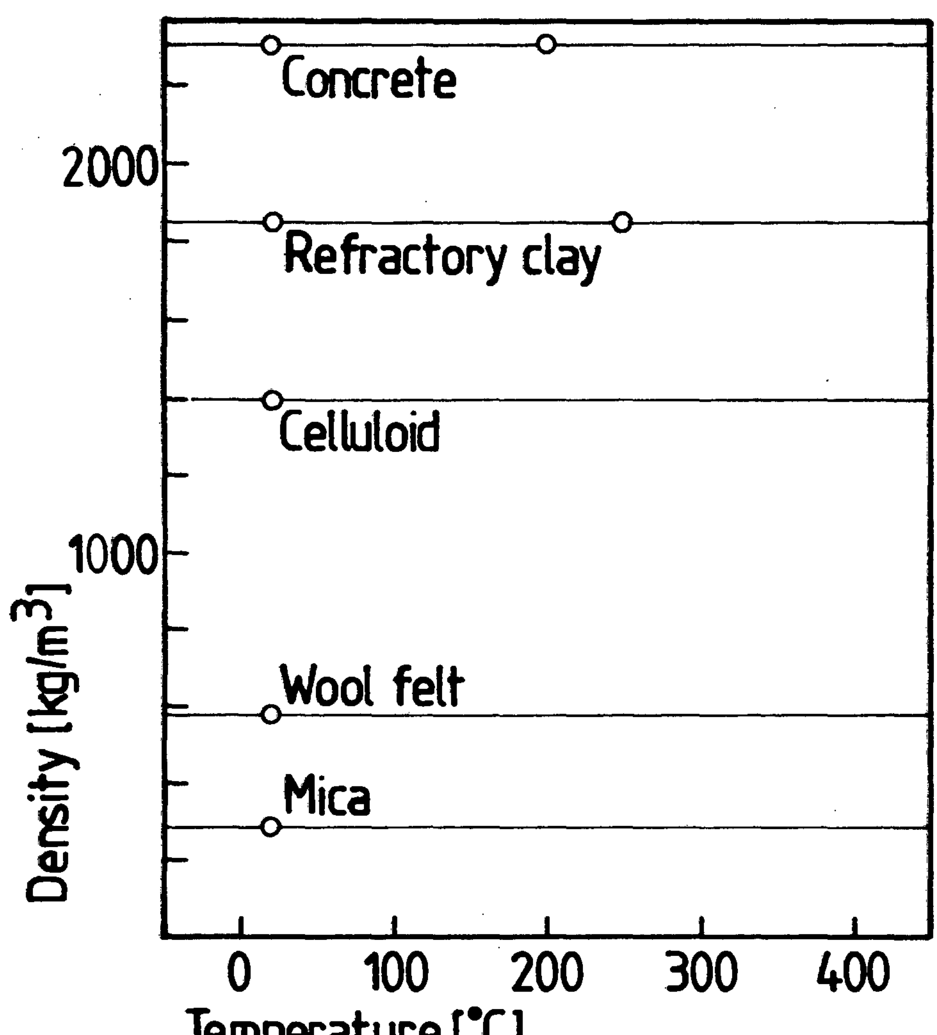


Figure 4.2: Density variation for some materials

SPECIFIC HEAT

The specific heat of a substance is a measure of the variation of its stored energy with temperature. In thermodynamics we recognise two specific heats:

specific heat at constant pressure: $C_p = \partial h/\partial\theta \big|_{p\ const}$

specific heat at constant volume: $C_v = \partial u/\partial\theta \big|_{v\ const}$, [4.2]

where $h = h(\theta,p)$ is the enthalpy per unit mass, and

$u = u(\theta,v)$ is the internal energy per unit mass.

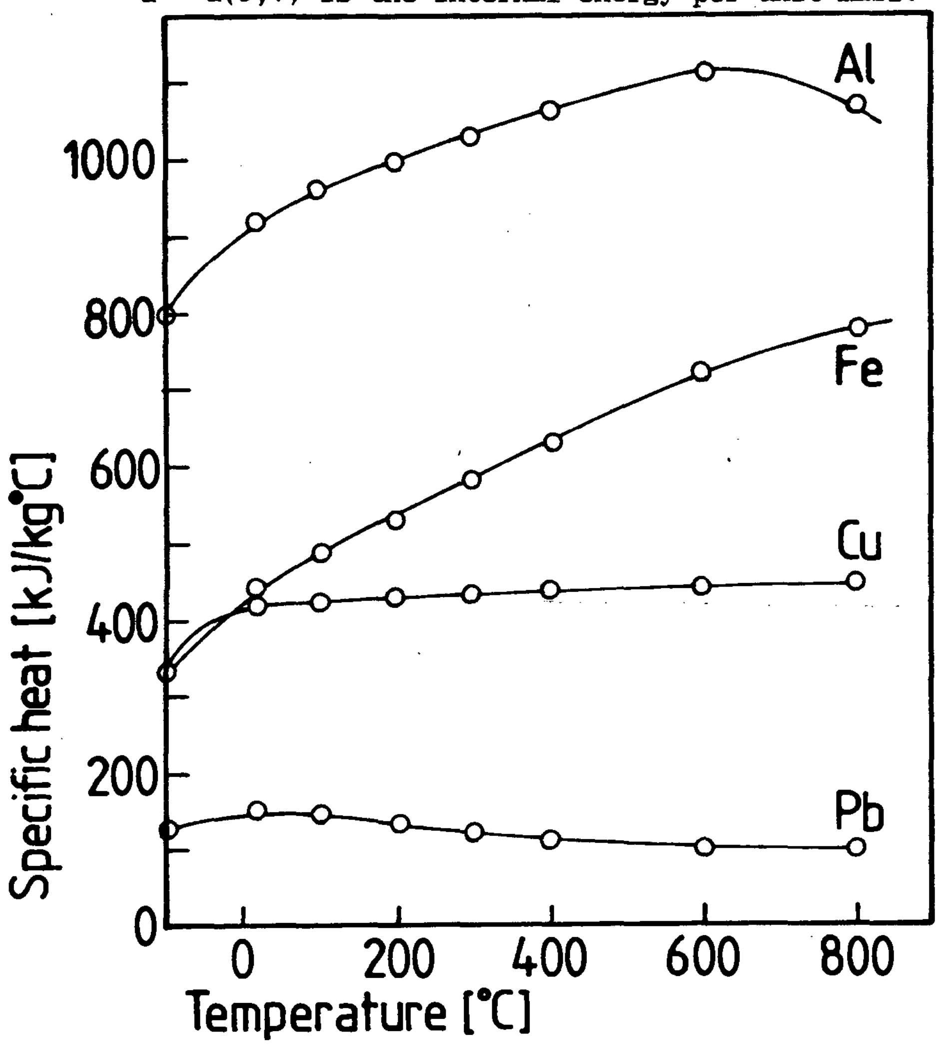


Figure 4.3: Specific heats for some materials

For nearly incompressible materials, such as solids and liquids, C_p is numerically equal to C_v. The specific heat of most materials is very dependent on the temperature of the body, and C_p is independent of pressure. This is demonstrated in figure 4.3. The above is not true for gases, but holds for liquids. Linear approximations of C_p are sufficient to ensure accuracy of results.

THERMAL CONDUCTIVITY

Thermal conductivity is a physical property which indicates the effectiveness of a substance in transferring heat by conduction. Its value is equal to the heat transferred per unit time, through a unit area, under a unit temperature gradient.

The dependence of k on temperature is very slight (weak) for solids. This is illustrated for a number of materials in figure 4.4. The importance of including this nonlinearity can be demonstrated by considering the slope of the curve describing the variation of k with temperature.

Consider a simple wall with one side held at zero and the other side at unit temperature. Allowing k to take on three possible forms, we plot the variation of the temperature through the wall, shown in figure 4.5.

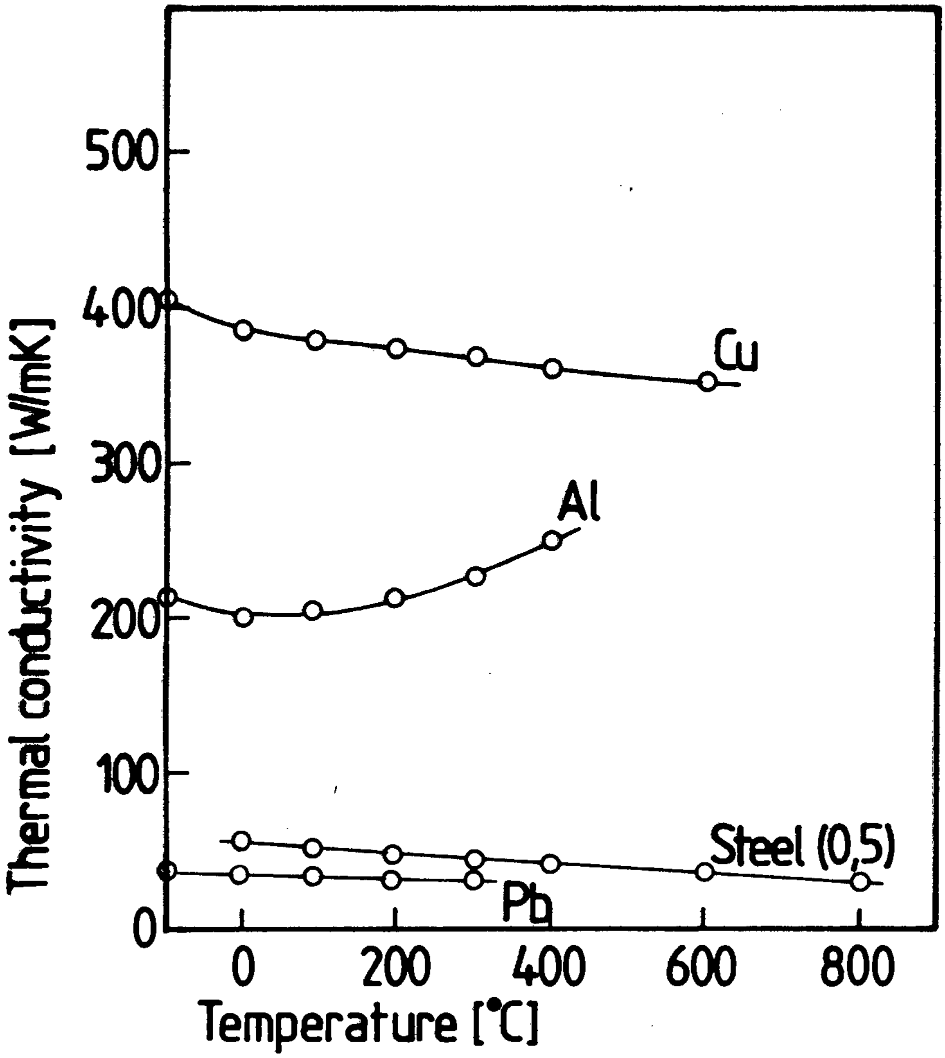


Figure 4.4: Thermal conductivities for some materials

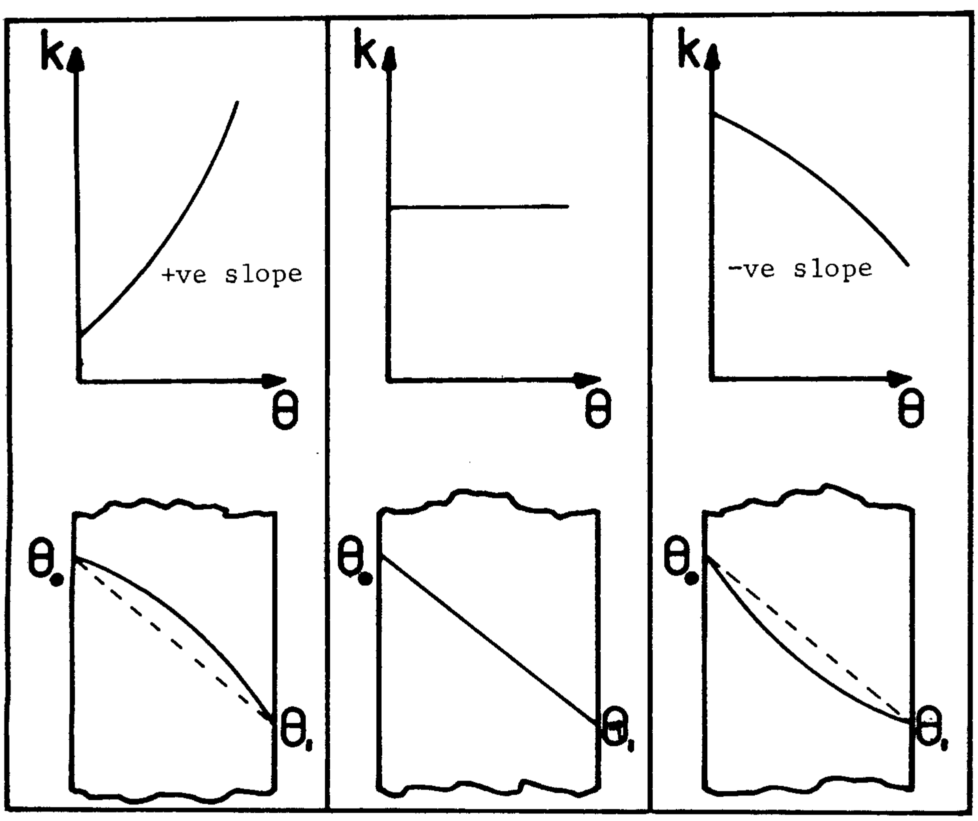


Figure 4.5: Effect of k(θ) on temperature distribution

Clearly, considering the temperature at the centre of the wall, it is very important to include variations in k.

The thermal conductivity of solids varies from values as low as those known for gases to values several orders of magnitude higher, as shown in table 4.1.

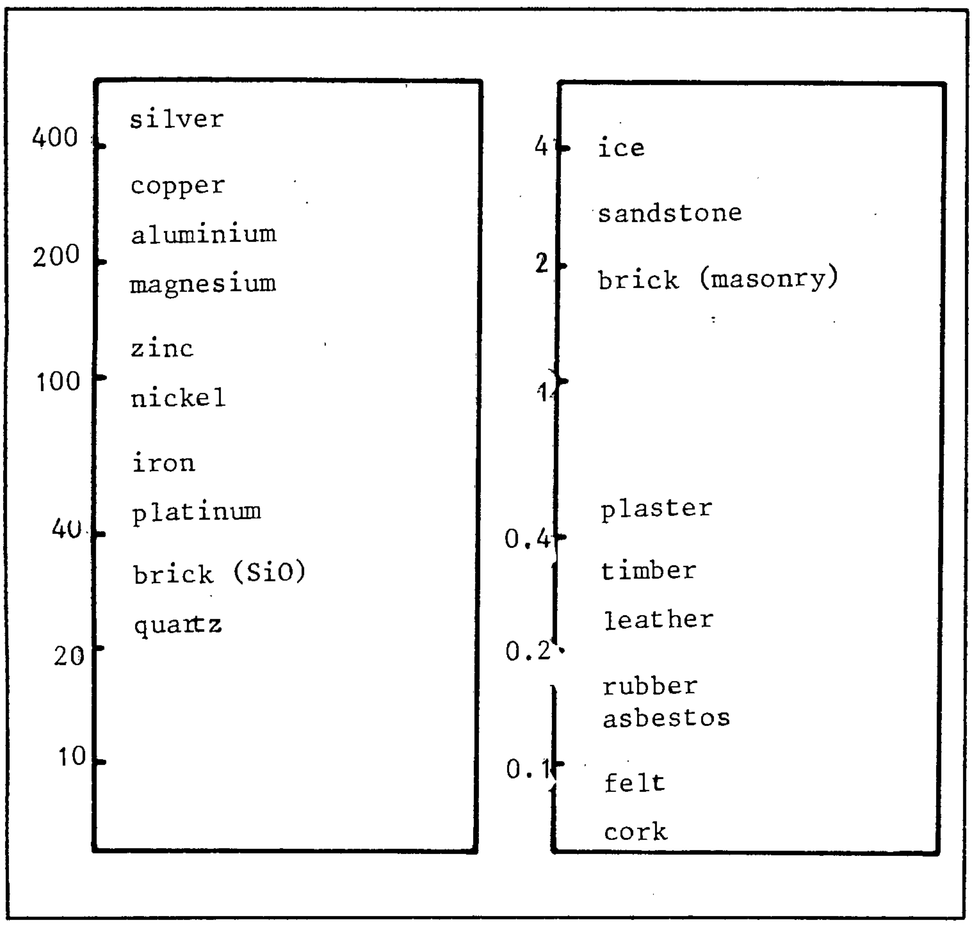


Table 4.1: The range of thermal conductivities for solids

The high values of thermal conductivity are observed for metals, especially in metals of high electrical conductivity. The reason for this is that heat flow can take place through one of two (known) mechanisms; lattice vibration and the movement of free electrons. Note from figure 4.6 that the thermal conductivities of insulating materials take on simple linear functions with temperature, probably due to the lack of free electrons in the materials.

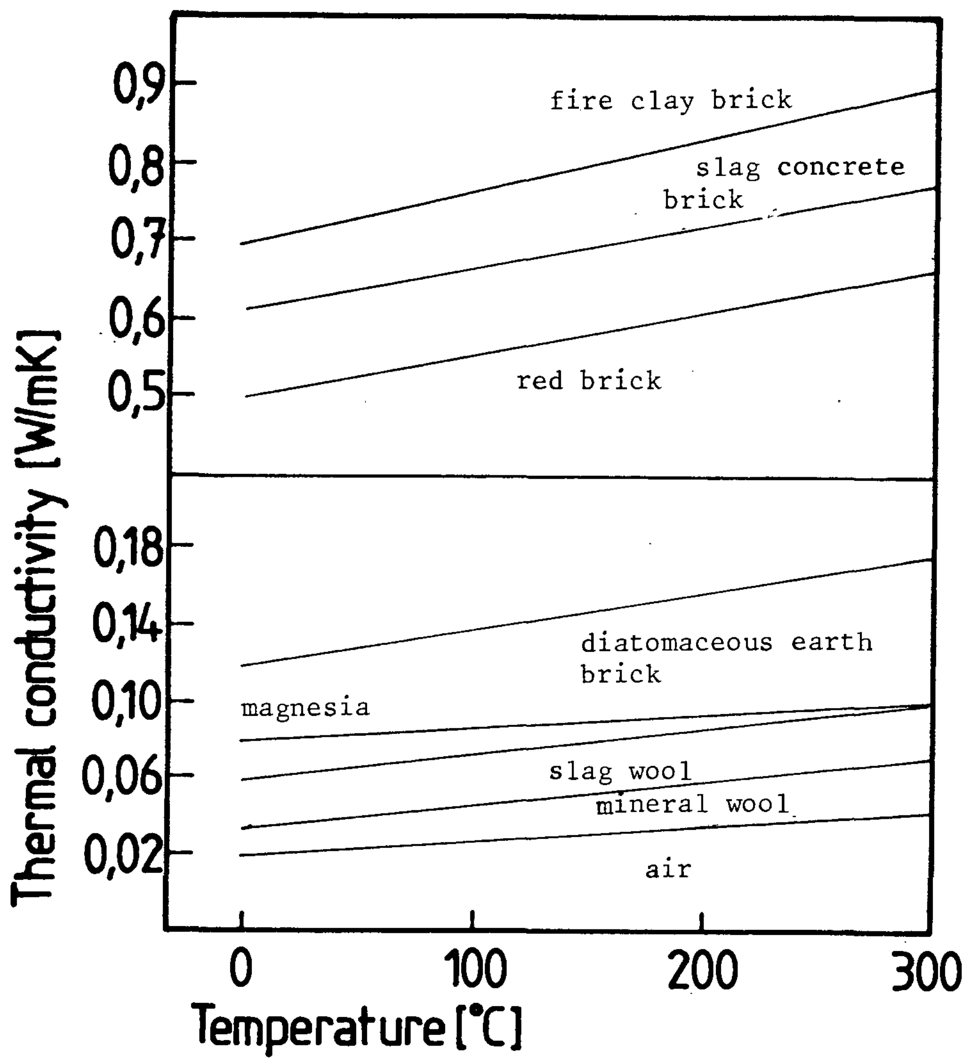


Figure 4.6: Thermal conductivities for some insulating and refractory materials

CONVECTION COEFFICIENT

The convective heat transfer coefficient h_c is defined as the constant of proportionality relating the heat transfer per unit time and unit area to the overall temperature difference between the solid and the liquid in contact with the solid. This relationship comes from Newton's law of cooling, equation [2.5c].

There exists no simple law describing the behaviour of h_c, as it depends on numerous factors, such as the quality of contact between the fluid and solid, the state of motion of the fluid, as well as the absolute temperatures of both the solid and liquid. Most texts give at least ten empirical relations, all with limited degrees of validity.

RADIATION COEFFICIENT

This coefficient is governed by a number of factors. Of the incident radiation, some is transmitted, some reflected and the rest is absorbed by the body. The amounts in each of the above cases vary with the wavelength of the incident radiation. Furthermore, the body radiates to the surrounding space depending on the relative temperature difference to the fourth power, the view factor, a geometrical factor which relates the effective 'visibility' between the body and the surrounding space.

To evaluate this coefficient numerically would require an extensive expert system which would have to make decisions based on the geometry of the body and the configuration of the surrounding space, as well as the surface quality of all participating objects. Fortunately this is not too much of a problem, as the major nonlinearity is due to the fourth order temperature terms.

INTERNAL HEAT GENERATION FUNCTION

Systems with heat sources and sinks are often found, and include electric coils, resistance heaters, exo- and endothermic reactions (such as fuel combustion or the liberation of ammonia from water), as well as nuclear reactions.

Fortunately, many of these functions are previously known, and as a result are included in the formalism simply as a thermally varying polynomial. The value can also be coupled to a time function, which is also a polynomial.

4.2 Piecewise linear interpolation

In the discussion that follows, reference will only be made to thermal conductivity. Obviously it is equally valid for any material property.

It has already been mentioned that the thermal conductivity varies with temperature. For numerous materials (as can be seen from the preceding illustrations), especially within a limited temperature range, the variation of thermal conductivity with the temperature can be represented by the linear function

$$k(\theta) = k_o(1 + a_o\theta), \qquad [4.3]$$

where k_o is the thermal conductivity at $\theta=0$, and

a_o is the temperature coefficient of thermal conductivity.

Assuming that we have a thermal conductivity variation as represented by figure 4.7, and that we wish to analyse a problem for the range θ_1 to θ_2.

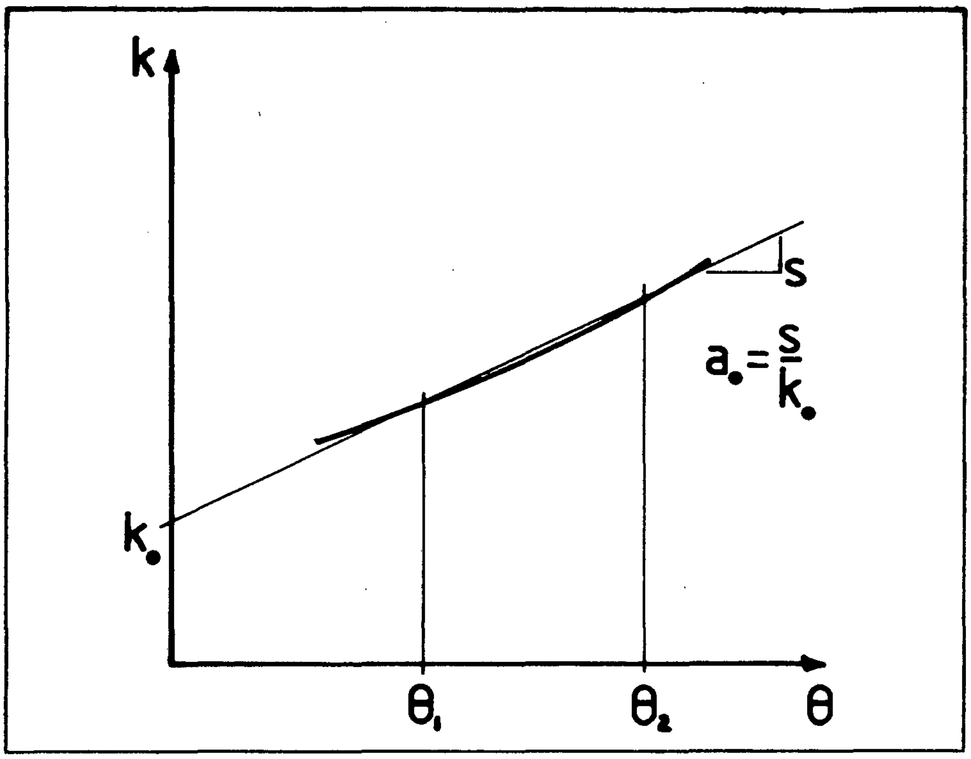


Figure 4.7: Linear approximation for a small range of k

We would proceed by constructing the best straight line fit in that range, and extend the line to cut the vertical axis. The value of k_o will then be the vertical axis intercept, and the value of a_o will be the slope of the line divided by k_o.

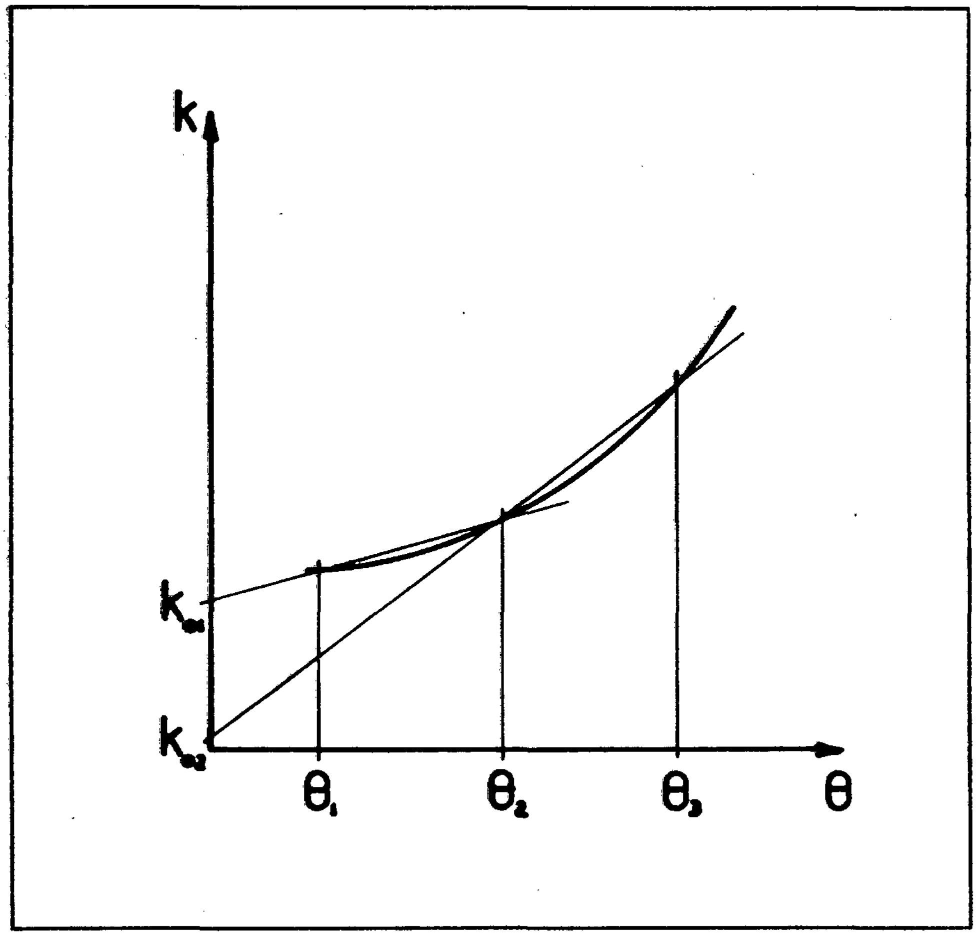


Figure 4.8: Piecewise linear approximation for the entire range

For some problems a single straight line approximation may not be sufficiently accurate. In such a case it is possible to extend the number of linear interpolation functions, so that:

$$k = k_{o1}(1 + a_{o1}\theta) \qquad \theta_0<\theta<\theta_1$$
$$k = k_{o2}(1 + a_{o2}\theta) \qquad \theta_1<\theta<\theta_2$$
$$\cdot \qquad \cdot \qquad \cdot$$
$$k = k_{on}(1 + a_{on}\theta) \qquad \theta_{n-1}<\theta<\theta_n \qquad [4.4]$$

This is demonstrated in figure 4.8. It is thus possible to approximate an arbitrary function to within reasonable accuracy using piecewise linear interpolation functions.

CHAPTER FIVE: NUMERICAL IMPLEMENTATION

5.1 Introduction

This chapter is concerned with the numerical implementation of the transient heat conduction problem. The preceding theory is only useful if it is presented in a form in which non-trivial problems can be solved. This implies using a fast numerical machine with large storage capabilities. For this reason a Sperry Univac mainframe was used to install the program. The routines which were written to implement the transient heat conduction equations form part of the program NOSTRUM (NOnlinear STRuctural Mechanics), which is under constant development at the Applied Mechanics Research Unit at the University of Cape Town [24].

Although NOSTRUM possesses a steady-state thermal analysis routine, the idea is to eventually have a fully coupled thermo-mechanical capability. This is only possible using a transient thermal analysis formulation. To enable the thermal analysis facility to be incorporated where needed at a later stage (for example within a mechanical analysis program), an independent driver routine is employed. This routine is called directly from the NOSTRUM driver routine.

5.2 Program structure

The layout of NOSTRUM as regards variable name conventions and loop structures has been followed closely for consistency. The present program has been designed in a modular way, so that the addition of new features is a simple matter. In some cases, these features have been anticipated, and empty (shell) routines have been programmed, so that only the details need be added. These shells are indicated in grey in the flowcharts that follow. As many existing routines as possible were utilised to ease the integration of the program into NOSTRUM.

The NOSTRUM modules for initialising arrays and variables, and the data input and checking routines were simply altered to incorporate the transient thermal analysis information. This was done by extending the use of existing flags in the input routine to include the additional options.

It is important to take note of the following information when considering the program code. The program was written in ASCII FORTRAN, firstly because NOSTRUM is written in the same language, and secondly because ASCII FORTRAN has many structured commands, and will always form a subset of any upgrades of the language. Finally, FORTRAN is still considered one of the most portable computer languages, and hence a change in hardware should only minimally affect the source code. Insofar as the program is concerned the formulation requires only one degree of freedom per node, the temperature, which alters the appearance of the gradient and shape function matrices from that of the mechanical formulation. Presently only plane isoparametric elements are available for thermal analysis. The code solves for the total state variable, hence the frontal solver has had to be adapted, and does not correspond to the usual NOSTRUM solver.

Figure 5.1 gives the overall structure of the program, single lines indicating program sequence, and double lines indicating program loops. Subsequent flowdiagrams describe the details of each of the programs referred to in the structure chart depicted in figure 5.1. Table 5.1 briefly describes the functions of each of the subroutines found in the structure chart of figure 5.1.

5.3 Model input

The model input is devided into a number of stages. The master control parameters define the type of problem, and here a flag is set that will cause the necessary data from the subsequent input data stream to be filtered out. Included in these parameters is the solution control, the time step length, problem duration and the time integration parameter α. The next control parameters to be entered are those that define the heat transfer problem. At this stage one can decide to include effects due to convection, radiation, internal heat generation and point sources. The program also expects the initial nodal values of temperature. If none are specified, all nodal values will be set to zero. For the model geometry data no change to the input routine has been necessary, and all existing data structures have been used as is. These include the connectivity matrix, the local and global node numbers as well as the local and global coordinates. The next step is to enter the properties associated with each element group and its boundary. Properties may be defined as either constant, linearly dependent on temperature, or a piecewise linear function may be defined over a temperature range. The thermal conductivity may be specified to be orthotropic or isotropic, whereas the other parameters are all assumed to be scalars. Finally the model loading has to be specified. Each load is entered with an associated time function which will specify the magnitude of the load at a particular time instant of the analysis. The loading options presently available are:

1. On the Dirichlet boundary the temperature must be specified.

2. On the Neumann boundary the flux may either be specified to take on some value, or be set to zero (adiabatic condition). A zero flux is useful for reducing the size of symmetrical problems, as this is the condition on a line of symmetry.

For more information on the model input stage the users manual is recommended.

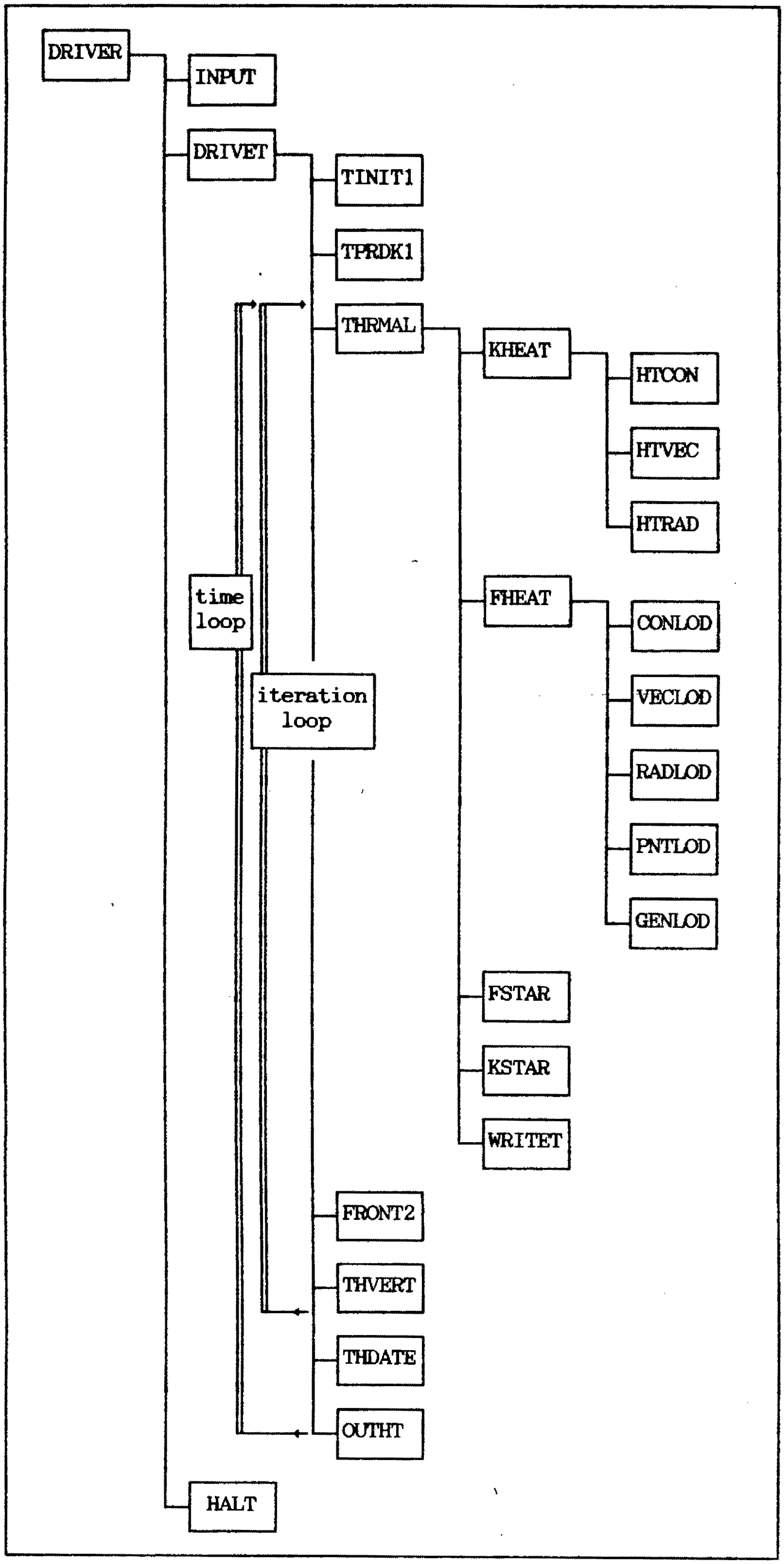


Figure 5.1: Structure chart of the program

ROUTINE	FUNCTION
DRIVER	The main general purpose NOSTRUM driver routine which controls the solution processes for all nonlinear problems.
INPUT	The general purpose processor of NOSTRUM for entering the user defined problem input data deck
DRIVET	Driver to control the solution processes of transient first order parabolic type problems.
TINIT1	Routine to initialise the temperature vector with either user or program starting conditions.
TPRDK1	Routine to predict the temperatures and their rates for a new time step.
THRMAL	Assembly of element matrices and vectors. Routine loops over element groups, and elements in group.
KHEAT	Routine to control the assembly of the element conductivity matrix.
HTCON.. ..HTRAD	Routines assemble contribution to conductivity matrix due to conduction....radiation.
FHEAT	Routine to control the assembly of the element heat load vector.
CONLOD.. ..GENLOD	Routines assemble contribution to load vector due to conduction....internal heat generation.
KSTAR	Routine to calculate the effective 'stiffness' matrix for the current element.
FSTAR	Routine to calculate the effective 'load, vector for the current element.
WRITET	Routine writes the effective 'stiffness' matrix to disk storage for use by frontal solver.
FRONT2	Frontal solver for one degree of freedom per node, non incremental state variable.
THVERT	Routine to check for the convergence of two consecutive iterations.
THUPD	Routine to update the state variable vectors for a next iteration.
RESET1	Routine to reset the state variable vectors for the next time step.
OUTHT	Routine to print out nodal information at required time intervals.
WARN	Routine to handle error levels and messages.

Table 5.1: Summary of Program functions.

5.4 Assembling the element coefficient matrices and vectors

The equations which are to be evaluated by the program to obtain the coefficient arrays have been derived in chapter 2 and are summarised by equation [2.43]. In the case of matrices being evaluated, computational effort is saved by only calculating the upper triangle. The lower half is then copied over, which has a further advantage in that the matrices are forced to be exactly symmetrical.

The Conductivity Matrix.

The conductivity matrix is often referred to as the stiffness matrix in the literature as it is of a similar form. The general conductivity matrix is made up of three components, namely conduction, radiation and convection contributions. For the purposes of this thesis only the contributions due to conduction will be considered.

The thermal conduction matrix corresponds to the elasticity matrix, and in the present formulation it is set as a diagonal matrix, which implies that only isotropic and orthotropic materials may be modelled. The algorithm for the assembly of the matrix is now given.

For each element:

1.Set up integration sampling points and weights.

2.Determine element geometry and global node numbers.

3.Determine the nodal temperatures from the previous time step.

4.Determine the material properties for the element.

5.Initialise the conductivity matrix.

6.For each integration point:

6.1.Determine the current sampling position.

6.2.Determine the shape functions and derivatives for this point.

6.3.Calculate the value of the Jacobian derivative at the present sampling point.

6.4.Calculate the sampling point temperature.

6.5.Evaluate the conduction matrix at the sampling point.

6.6.Evaluate the contribution of $[B^T][k][B]|J|$ for the present sampling point. Add this to the conductivity matrix.

The Specific Heat Matrix.

The specific heat is often referred to as the damping matrix as it is associated with the first time derivative of the state-variable. It is of a similar form to the conduction matrix, but somewhat simpler to assemble as it only requires the evaluation of a scalar for the material property and requires the shape functions instead of their derivatives. The following algorithm describes the assembly.

For each element:

1. Set up integration sampling points and weights.

2. Determine element geometry and gobal node numbers.

3. Determine the nodal temperatures from the previous time step.

4. Determine the material properties for the element.

5. Initialise the capacitance matrix.

6. For each integration point:

   6.1. Determine the current sampling position.

   6.2. Determine the shape functions for this point.

   6.3. Calculate the value of the Jacobian derivative at the present sampling point.

   6.4. Calculate the sampling point temperature.

   6.5. Evaluate the specific heat at the sampling point.

   6.6. Evaluate the contribution of $C_p \rho [N^T][N]|J|$ for the present sampling point. Add this to the capacitance matrix.

The Heat Load Vector.

Since it is desirable to include load variations on the boundaries it is not sufficient to merely specify point loads for each element. We need to calculate the consistent thermal load vector which will naturally take care of variations across the boundary. Contributions to the load vector may come from convection and radiation boundaries, but for the purposes of this thesis these contributions have been omitted. Only specified flux loading will be considered.

The major distinction between the preceding matrices and the load vector is that they are evaluated over the volume of the element, whereas the load vector is calculated over the element boundary. Since we wish to maintain the generality of our formalism by only integrating over the master element, we need to adjust the Jacobian determinant for the boundary. This is either accomplished by setting one of the sampling positions and its associated weight equal to unity, and to then sum over the four boundaries, or else one can explicity derive the transformation and hence avoid evaluating the jacobian determinant. The latter approach has been followed in the thesis. The assembly is described in the algorithm below.

For each element:

1. Determine whether the element is loaded, and on which sides.
2. For each loaded side
    1. Calculate the consistent nodal flux values.
    2. Determine the Integration points and sample positions.
    3. For each integration point:
        1. Determine the element geometry and node numbers.
        2. Determine the material properties.
        3. Determine the present integration point position.
        4. Determine the shape functions at the integration points.
        5. Evaluate the change of coordinate system derivatives $\partial x/\partial \xi$ and $\partial y/\partial \xi$ at the sampling point.
        6. Calculate the contribution due to this integration point, and add it to the load vector.

The Effective Arrays.

The system of equations is linearised by forming an effective stiffness matrix and an effective load vector. These are evaluated for each element separately, and are then stored away to diskfile for subsequent retrieval by the frontal solver. To calculate these arrays, the time integration parameter α and the time step length $\Delta\tau$ need to be given. The details of the calculation are again given by an algorithm.

For each node i per element:

1.For each node j per element:

1.Evaluate: $[K_{eff}(i,j)] = [K(i,j)] + \frac{1}{\alpha\,\Delta\tau}[C(i,j)]$

For each node i per element:

1.For each node j per element:

1.Evaluate:

$$[f_1(i)] = [f_1(i)] + [C(i,j)]\{\frac{1}{\alpha\,\Delta\tau}\,\theta(j) + \frac{(1-\alpha)}{\alpha}\,\dot{\theta}(j)\}$$

2.Evaluate:

$$[f_{eff}(i)] = [f(i)] + [f_1(i)]$$

The Transient Driver.

The transient nonlinear matrix equation has been linearised which makes it simple to solve, but to get it in this form we require a routine to keep track of the various time step and iteration solutions, as well as to do the housekeeing of updating the relevant arrays. This routine has effective control over the solution process, from the time stepping and prediction of variables, to the iterative procedure to reach a stable solution.

Initialise θ_{n-1}.

Predict: $\theta_n^i, \dot{\theta}_n^i$

For time step n:

1.For iteration i:

1.1.Determine: $[K(\theta_n^i)], [C(\theta_n^i)], [f(\theta_n^i)]$

1.2.Determine: $[K_{eff}], [f_{eff}]$

1.3.Solve: $[K_{eff}]\theta_n^{i+1} = [f_{eff}]$

1.4.Check: is $|\theta_n^{i+1} - \theta_n^i| <$ tolerance ?

1.4.1.No: $\theta_n^i = \theta_n^{i+1}$, next iteration.

1.4.2.Yes:

1.4.2.1.Determine: $\dot{\theta}_n^{i+1} = \frac{1}{\alpha\,\Delta\tau}(\theta_n^{i+1} - \theta_{n-1}) + \frac{1-\alpha}{\alpha}\,\dot{\theta}_{n-1}$

1.4.2.2.Reset: $\theta_{n-1} = \theta_n^{i+1}$

$\dot{\theta}_{n-1} = \dot{\theta}_n^{i+1}$

1.4.2.3.Predict: $\dot{\theta}_n^i = \dot{\theta}_{n-1}$

$\theta_n^i = \theta_{n-1} + \Delta\tau\dot{\theta}_n^i$

1.4.2.4.Next time step.

CHAPTER SIX: ILLUSTRATIVE EXAMPLES

The examples listed in the body of this chapter have been chosen to test the validity of the formulation presented in the thesis with analytical results, as well as to enable one to gauge the stability of the numerical scheme. The first two examples are from the verification manual of the ADINAT program, and the third example from the standard text by Carslaw and Jaeger [1]. The analytical results were obtained from the original sources. Finally, a simple problem was designed to show some of the features of the formalism.

6.1 Problem 1: Transient conduction in a semi-infinite wall.

Problem:

An homogenous semi-infinite solid is initially at zero temperature when the surface is subjected to a unit step forcing function at time $t = 0^{+}$. It is required to find the time history of temperature variation through the body. The problem is depicted in figure 6.1.

Model:

Fifteen similar elements of length 0.2 and area 1.0 were chosen to represent the problem. Both four-noded and eight-noded elements were used, to allow for comparison of results.

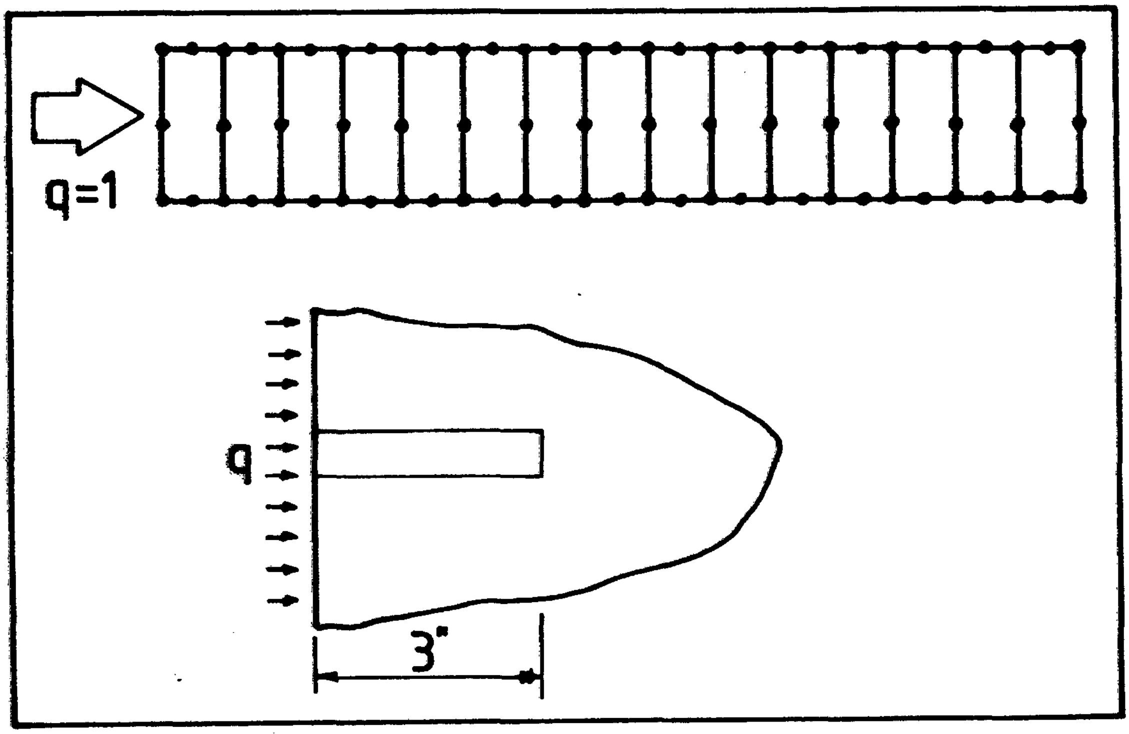


Figure 6.1: Physical problem and finite element model.

Results:

The results for the surface temperature were plotted and are shown in figure 6.2. Reasonable correspondence with the analytical solution was obtained, but only for time step lengths of less than 0.01 seconds. This is somewhat less than the 0.025 second time step reported in the Adinat verification manual [25]. This may be due to the difference between the time integration algorithm adopted by the Adina Engineering group and that adopted in the present formulation. The analytical solution is due to Carslaw and Jaeger [1].

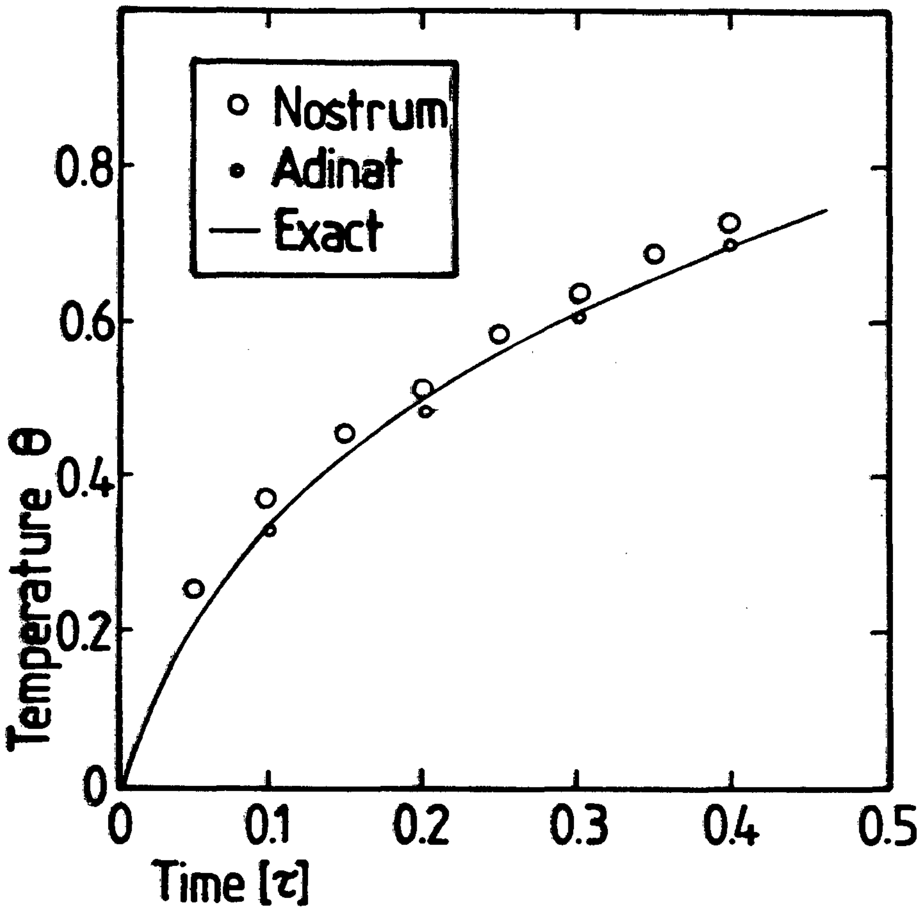


Figure 6.2: Time history of surface temperature.

6.2 Problem 2: Conduction in a square column.

Problem:

A square column is subjected to prescribed temperature boundary conditions, with three surfaces held at zero temperature and the fourth at one hundred degrees Fahrenheit. It is required to find the final steady-state temperature distribution in the column.

Model:

The column was modelled using eight square eight-noded elements, and reduced in size by exploiting the symmetry of the problem.

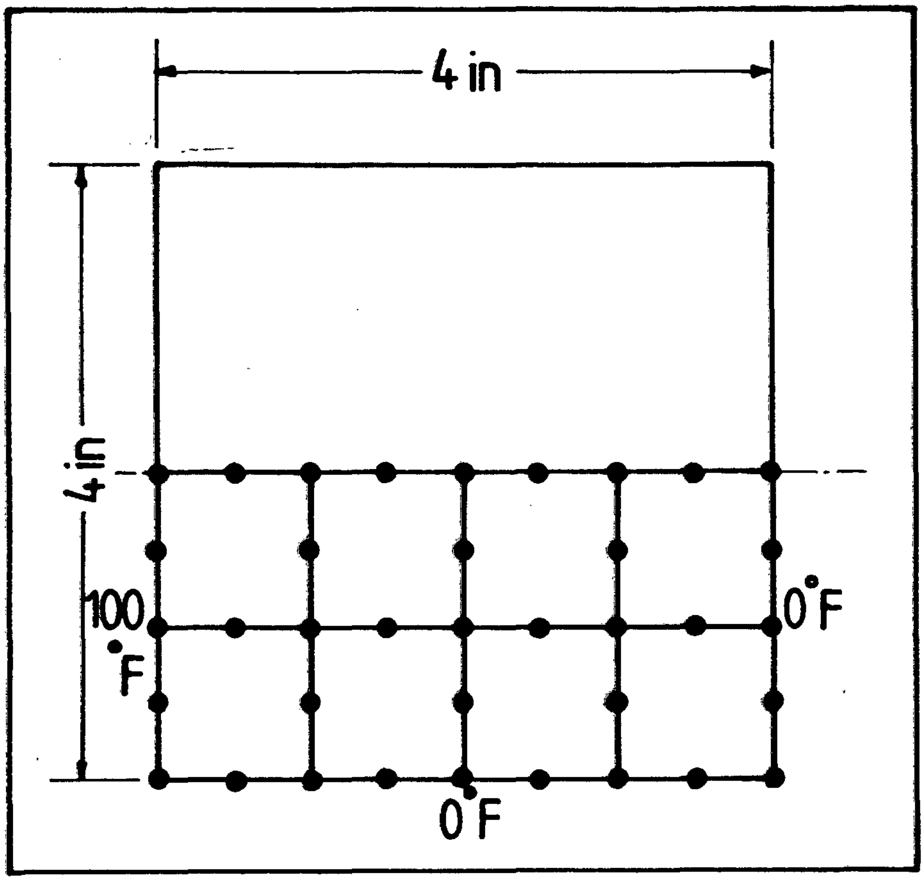


Figure 6.3: Physical problem and finite element model.

Results:

The results for the test were plotted against the analytical solution and is shown in figure 6.4. Results were obtained in two ways, both yielding exact correspondence with the analytical solution. Firstly a transient solution was initialised which reached the steady state solution after 16 seconds. The same problem was modelled again, but with the specific heat set to zero. This immediately yielded the steady-state solution.

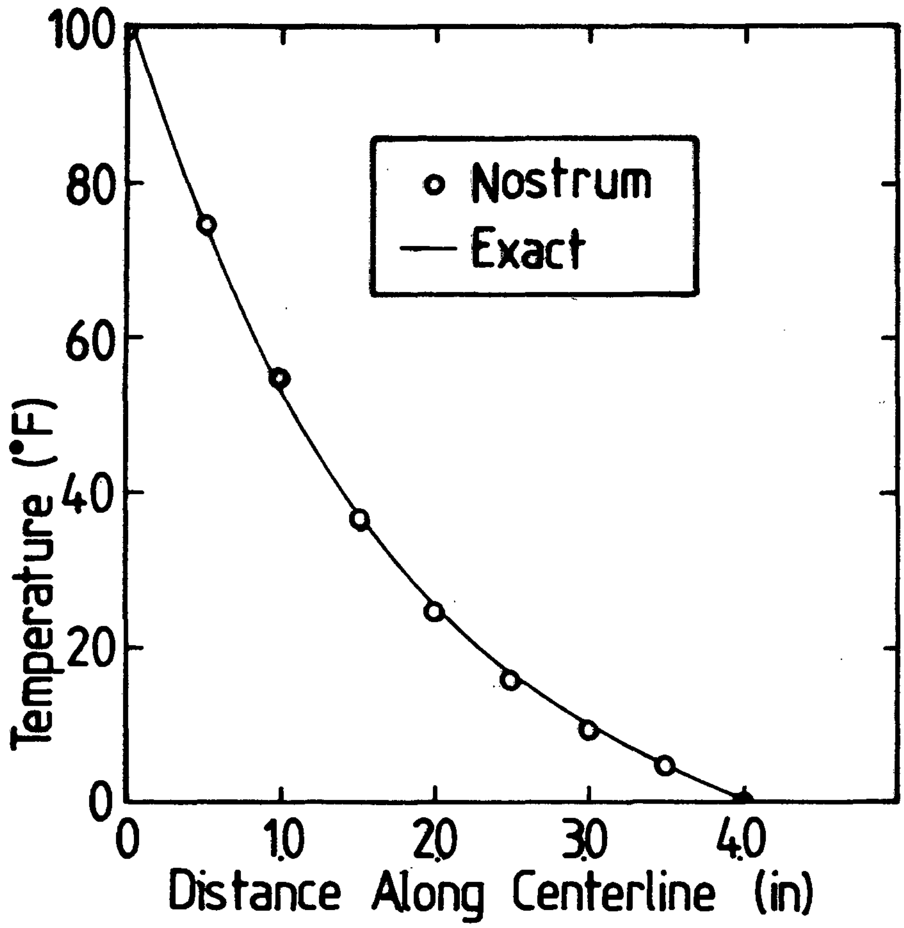


Figure 6.4: Temperature distribution in a square column.

6.3 Problem 3: Initially imposed thermal field.

Problem:

A wall of thickness 2l, initially at zero temperature, has both boundaries suddenly raised to unit temperature. It is required to find the time history of the temperature in the wall. The problem may be simplified by exploiting symmetry. The model is shown in figure 6.5

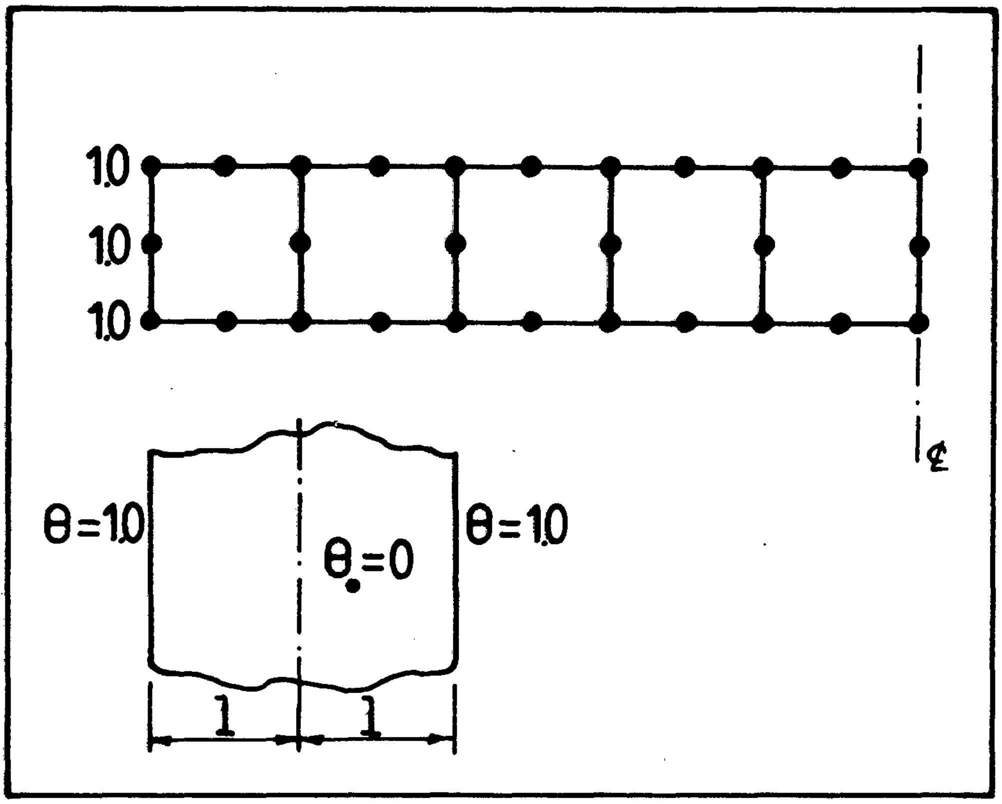


Figure 6.5: Physical problem and finite element model.

Results:

Carslaw and Jaeger [36] give the following Fourier series solution for this problem:

$$\frac{\theta}{\theta_o} = 1 - \frac{4}{\pi}\sum_{n=0}^{\infty}\frac{(-1)^n}{2n+1}\, e^{-(2n+1)^2\pi^2 T/4}\cos\frac{(2n-1)\pi\xi}{2},$$

where

$T = \frac{\rho\tau}{kl^2}$ and $\xi = \frac{x}{l}$, the dimensionless temperature and distance coordinates. When we compare the finite element results with the above solution, we obtain almost exact correspondence. This is shown in figure 6.6.

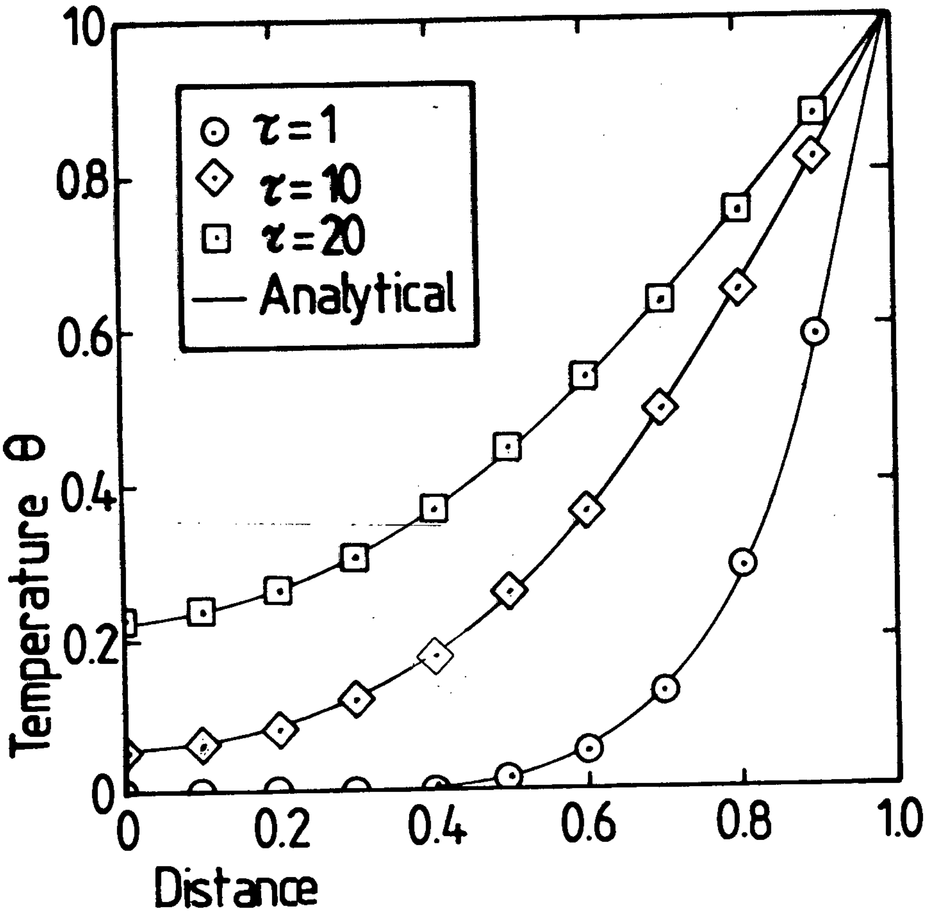


Figure 6.6: Transient response of the wall.

6.4 Problem 4: Transient conduction in a simple slab.

This problem was designed to test some of the transient features of the formalism.

Problem:

A semi-infinite solid is subjected to a periodic pulse on its surface. It is required to study the behaviour of the thermal response at some interior point. A point one unit inside the solid was selected.

Model:

The finite element model of example 1 was employed, with the new boundary conditions imposed.

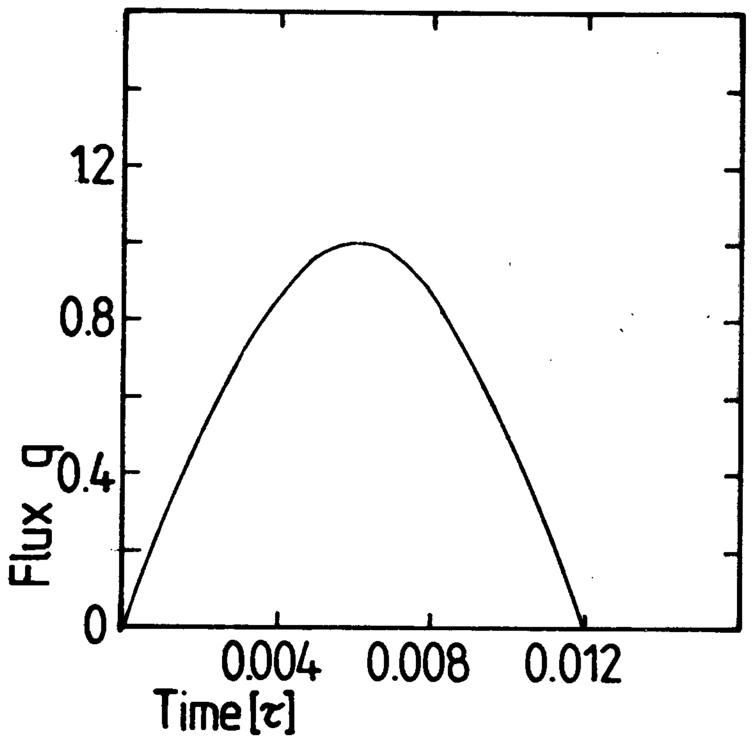


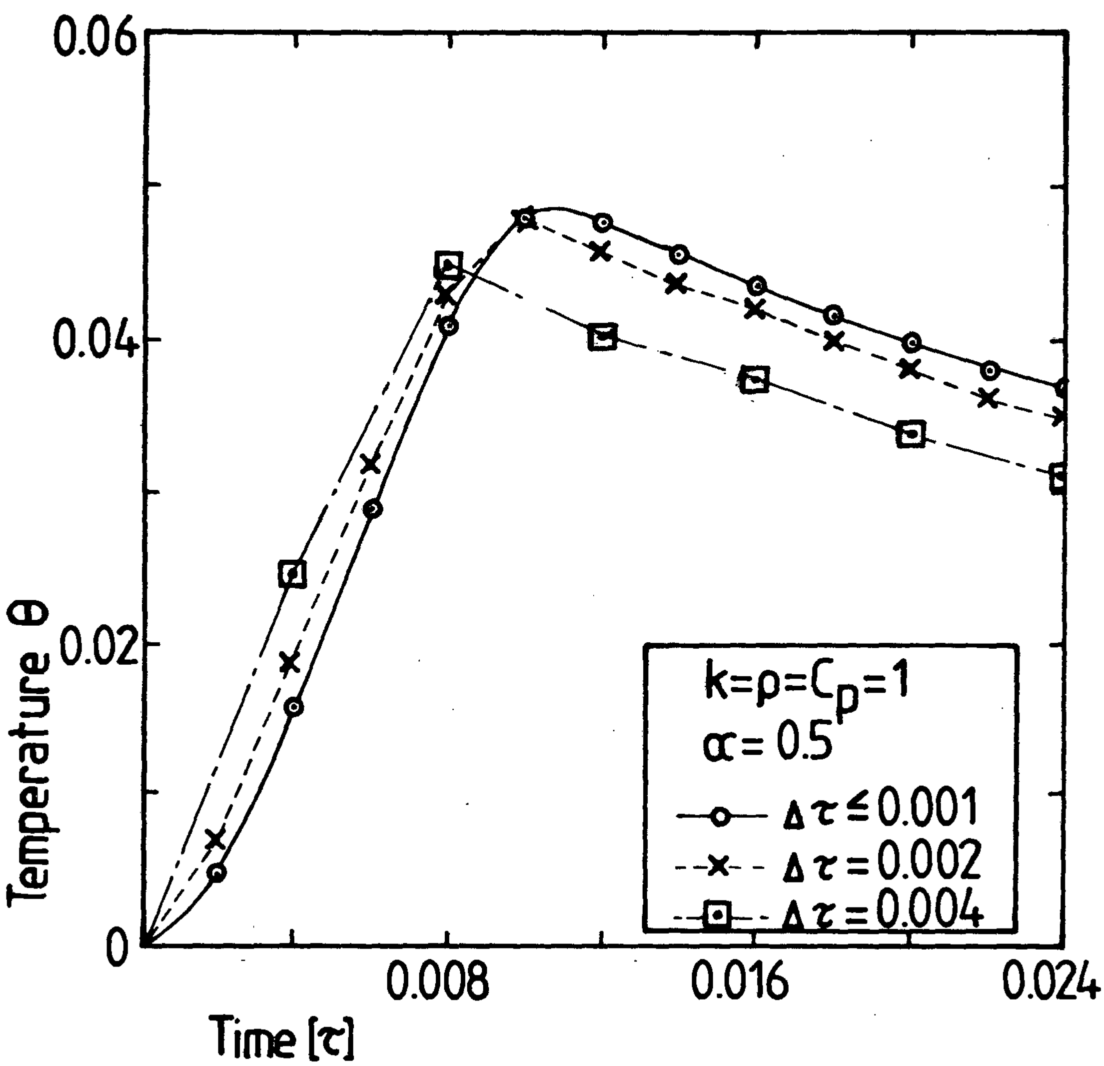
Figure 6.7: The sinusoidal pulse forcing function.

Results:

The time response for α = 0.75 was found for a number of time step lengths. The results converged to one solution for time steps less than or equal to 0.001. We will refer to this as the solution.

The maximum allowable time step length was found to be 0.01, beyond which the program yielded zero temperatures at all the nodes. This may be due to -ve pivots in the frontal solver.

The solution is followed fairly closely regardless of the choice of $\Delta\tau$ within the valid range. The time integration parameter α only has an effect on the solution at the first time steps (where the gradient of the solution path is very large), after which all the solutions converge and remain together for the rest of the solution. The results are plotted in figures 6.8 and 6.9.



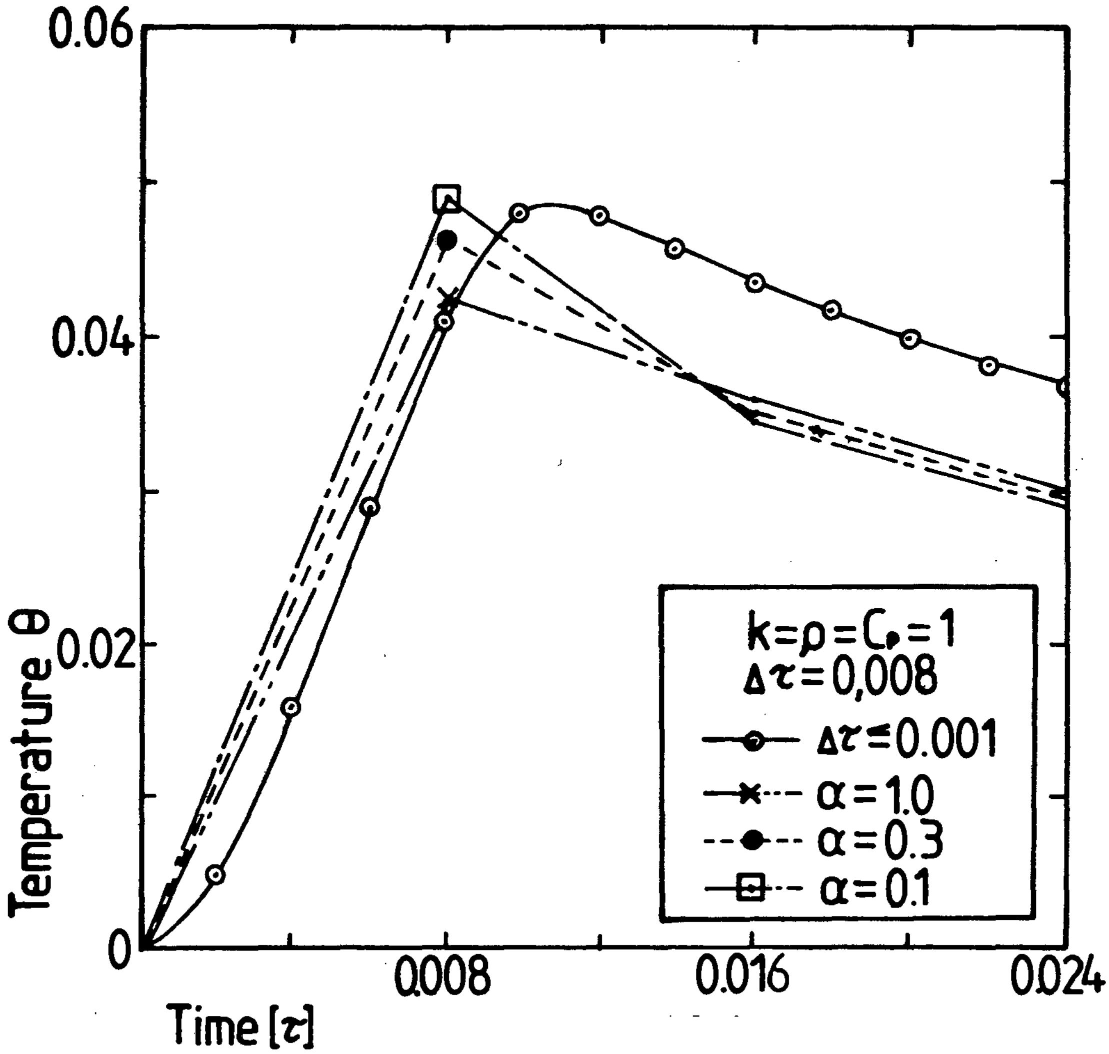


Figure 6.9: Results for various values of α.

CHAPTER SEVEN: CONCLUSIONS AND RECOMMENDATIONS

This thesis has considered the nonlinear transient heat conduction problem in solids. The problem was stated mathematically and the governing first order partial differential equation was derived. Because a finite element solution was sought, the functional of the partial differential equation was derived. Semi-discretisation of the functional in space was accomplished using the concepts of isoparametric elements and Gaussian quadrature. To discretise the equations in time, a direct integration method was followed. The integration scheme chosen has an unconditionally stable regime where the choice of time step length is governed only by accuracy requirements. This is a well-known feature of α-parameter integration methods, where $\alpha \geq 0.5$. The option of either an explicit or implicit time integration scheme still remains.

The fully discretised equations were implemented within the NOSTRUM finite element program of the Applied Mechanics Research Unit at the University of Cape Town. The assembly of the coefficient matrices were compared with analytical results and were found to be exact. Examples of these matrices are presented in Appendix A2. The numerical implementation was compared with analytical solutions from the literature. In all cases the steady-state results corresponded exactly. The program was further compared with the commercial finite element code ADINAT. Again almost exact correspondence of results was obtained. The examples presented in the thesis were selected to verify the numerical procedures, and as such do not show the full capabilities of the code.

To the future user it is recommended that transient analysis be performed with caution. A number of factors have to be considered carefully. When temperatures are very small, the time step large or the time integration parameter α around 0.5, slight numerical oscillations are produced. It is possible to damp the oscillations by reducing the time step length, or by increasing the parameter α to between 0.68 and 0.88. If the time step length is chosen too large the solution will drift away from the correct solution, usually with an increase in time constant in the case of a ramp response, or an increase in phase lag in the case of a periodic forcing function.

The present numerical implementation includes effects due to conduction, specific heat, specified thermal loads and specified initial thermal fields. The heat transfer code has not been coupled to the mechanical stess analysis routines. This would entail major alterations to the time stepping procedures, since the heat transfer analysis advances in real time, whereas the time variable in the mechanical program is an arbitrary parameter used merely to advance through the load cycle.

A number of refinements to the code are possible, and these are now recommended.

1. It is possible to alter the thermal driver routine such that only one time step is solved for, in which case it may easily be incorporated in the mechanical routine to solve for coupled problems. The fully coupled thermo-mechanical problem may then also be attempted by linking the heat generation terms of the heat transfer analysis to the mechanical heat dissipated.

2. The inclusion of more complex boundary conditions is recommended as these are more likely to form part of realistic engineering problems. These include convection and radiation from the body. It would be advantageous to allow for a user defined subroutine to evaluate the relevant properties involved, as in practise these are usually expressed as complicated empirical relations with limited validity.

3. A feature which would save considerable computational effort is that of dynamic time stepping, where the length of the next time step is governed by such factors as the number of iterations needed to reach the solution at the present time step, as well as the temperature rate of change at that instant. If the number of iterations are small, or the rate of change of θ is small, the initial prediction (before iterating to a solution) will be very close to the solution, and hence a longer time step is justified.

4. It is recommended that the thermal routine be given a separate data input module, linked to a data base. This preprocessor should perform all the data checking to eliminate unnecessary crashing of the analysis routines. This would facilitate setting

up problems and enable most errors to be eliminated before commencing the analysis stage.

5. The use of a general post-processing module for plotting temperature contours would be very useful. In the case of large or complicated geometries a graphic output would facilitate checking and interpretation of the results.

6. Finally, since the present program occupies less than 250 kilobytes of memory for both the variables and code, it is recommended that it be stripped of the mechanical routines and ported down to run on a micro or minicomputer. In the author's experience the convenience of a smaller dedicated machine outweighs the advantages of computational speed on a large computer, especially at the program development stage. Returning the program to the mainframe does not pose a major problem.

The analysis of transient heat transfer phenomena by approximate methods is complicated by the fact that few analytical solutions exist, and that experimental results are ambiguous due to the thermal inertia of recording equipment and the difficulty in isolating various contributing factors. This makes it difficult to gauge the 'quality' of a numerical solution easily. Hence it is important to follow standard guidelines in attempting a solution. The time step length should be at least one tenth of the smallest expected period of the solution so that the transient is not overstepped. Elements should be chosen to be as regular as possible and should be sized such that the element stiffness contributions are not unduly large or small. Failure to comply to this rule causes the global stiffness matrix to become over stiff up problems and enable most errors to be eliminated before commencing the analysis stage.

5. The use of a general post-processing module for plotting temperature contours would be very useful. In the case of large or complicated geometries a graphic output would facilitate checking and interpretation of the results.

6. Finally, since the present program occupies less than 250 kilobytes of memory for both the variables and code, it is recommended that it be stripped of the mechanical routines and ported down to run on a micro or minicomputer. In the author's experience the convenience of a smaller dedicated machine outweighs the advantages of computational speed on a large computer, especially at the program development stage. Returning the program to the mainframe does not pose a major problem.

The analysis of transient heat transfer phenomena by approximate methods is complicated by the fact that few analytical solutions exist, and that experimental results are ambiguous due to the thermal inertia of recording equipment and the difficulty in isolating various contributing factors. This makes it difficult to gauge the 'quality' of a numerical solution easily. Hence it is important to follow standard guidelines in attempting a solution. The time step length should be at least one tenth of the smallest expected period of the solution so that the transient is not overstepped. Elements should be chosen to be as regular as possible and should be sized such that the element stiffness contributions are not unduly large or small. Failure to comply to this rule causes the global stiffness matrix to become over stiff in some parts which weights the solution. Finally, a number of solutions should be attempted, each time varying the time step length to ensure that the numerical procedure has converged.

REFERENCES

1. CARSLAW,H.S. and JAEGER,J.C.; Conduction of heat in solids; 2^{nd} ed.; Oxford University Press; Oxford; 1959.

2. SURYANARAYANA,N.V.; Transient Response of Straight Fins; Journal of Heat Transfer, Proc. A.S.M.E.; August 1985, pp 417-423.

3. BATHE, K.J.; Finite Element Procedures in Engineering Analysis, Prentice-Hall; New Jersey; 1982.

4. ZIENKIEWICZ, O.C.; The finite element method in Engineering Science; McGraw-Hill; London; 1971.

5. BATHE,K-J, and WILSON,E.L.; Numerical Methods in Finite Element Analysis; Prentice-Hall; Englewood Cliffs; 1976.

6. ODEN,J.T. and REDDY,J.N.; An Introduction to the Mathematical

Theory of Finite Elements; John Wiley; New York; 1976.

7. ZIENKIEWICZ,O.C., and CHEUNG,Y.K.; Finite elements in the solution of field problems; The Engineer, pp507-510, Sept.24, 1964.

8. WILSON,E.L.,BATHE,K-J., and PETERSON,F.E.; Finite Element Analysis of Linear and Nonlinear Heat Transfer; J. Nuclear Engineering and Design; Vol. 29, pp 110-124; 1974.

9. ROHSENOW,W.M. and CHOI,W.Y.; Heat, Mass and Momentum Transfer; Prentice-Hall, Inc.; Englewood Cliffs, N.J.; 1961.

10. BERCOVIER,M. and ENGELMAN,M.; A Finite Element for the Numerical Solution of Viscous Incompressible Flow; J. Computational Physics; Vol 30, pp 181-201; 1979.

11. VISSER,W.; A Finite Element Method for the determination of non-stationary temperature distribution and thermal deformations; Proc. Conf. on Matrix Methods in Structural Mechanics; Wright-Patterson A.F. Base; Ohio; 1965.

12. WILSON,E.L. and NICKEL,R.E.; Application of the Finite

Element Method to Heat Conduction Analysis; J. Nuclear Engineering and Design; Vol. 4, pp 276-286; 1966.

13. WARD,J.C. and WILKS,J.; Second Sound and the Thermomechanical effect at very low temperatures; Phil. Mag., 43, pp48-50;1952.

14. DINGLE,R.B.; Derivation of the velocity of Second Sound from Maxwell's equation of transfer; Proc. Phys. Soc., A65,pp364-376, 1952.

15. CHANDRASEKHARAIAH, D.S.; Thermoelasticity with second sound: A review; Appl. Mech. Rev. (ASME); Vol 39, no 3, Mar 1986.

16. PENNINGTON, D.S.; Nonlinear Heat Transfer and Thermo/Mechanical Stress Analysis using Finite Elements; M.Sc Thesis, University of Cape Town; 1985.

17. WINGETT,J.M. and HUGHES,T.R.J.; Solution Algorithms for Nonlinear Transient Heat Conduction Analysis Employing Element-by-element Iterative Strategies; Computer Methods in Applied Mechanics and Engineering, 52(1985); Elsevier Science Publishers; North-Holland.

18. LUIKOV, A.V.; Analytical Heat Diffusion Theory; Academic Press, New York, 1968.

19. BIOT, M.A.; Variational Principles in Heat Transfer; Oxford University Press; Oxford; 1970.

20. HOLMAN, J.P.; Heat Transfer; McGraw-Hill; New York 1983; p 535 - 536.

21. BELYTSCHKO,T. and HUGHES,T.R.J.; Computational Methods for Transient Analysis (Vol 1); North-Holland; Amsterdam; 1983; Chapter 2.

22. DHATT,G. and TOUZOTT,G.; The Finite Element Method Displayed; John Wiley and Sons; New Delhi; 1984.

23. COOK,R.D.; Concepts and Applications of Finite Element Analysis; John Wiley and Sons; New York; 1981.

24. DUFFETT,G.A., GRIFFIN,T.B., MARTIN,J.B., MERCER,C.D., REDDY,B.D. and RESENDE,L.; NOSTRUM - A finite element program for nonlinear structural mehcanics; November 1983.

25. ADINAT Users Manual; ADINA Engineering; Watertown, Mass.; 1981.

26. REDDY, B.D.; Functional Analysis and Boundary Value Problems, to be published by Pitman; Cape Town; 1986.

27. BATHE,K-J. and KOSHGOFTAAR,M.R.; Finite Element Formulation and Solution of Nonlinear Heat Transfer; J. Nuclear Engineering and Design; Vol. 51, pp 389-401; 1979.

28. DAMJANIC,F. and OWEN,D.R.J.; Practical considerations for thermal transient finite element analysis using isoparametric elements; J. Nuclear Engineering and Design; Vol 69, pp 109-126; 1982.

29. OWEN,D.R.J. and DAMJANIC,F.; Reduced numerical integration in thermal transient finite element analysis; J. Computers & Structures; Vol 17, pp261-276; 1983.

30. JOHNSEN,A.E. and TOROK,D.; Software for fluid flow and heat transfer analyses of electronic packaging; J. Computers in Mechanical Engineering; July 1985, pp 41-46.

31. OWEN,D.R.J. and HINTON,E.; A Simple guide to Finite Elements; Pineridge Press; Swansea; 1980.

32. CONTE, S.D. and DE BOOR, C.; Elementary Numerical Analysis; McGraw -Hill, Kogakusha; 1982.

33. HILDEBRAND, F.B.; Advanced Calculus for Applications; Prentice/Hall International; New Jersey; 1962.

34. SMITH, J.M. and VAN NESS, H.C.; Introduction to Chemical Engineering Thermodynamics; McGraw-Hill,p. 114.; Kogakusha; 1982.

35. KELLY, K.K.; U.S. BUR. MINES. BULL. 584, 1960.

36. AMERICAN SOCIETY FOR METALS; Metals Handbook Vol. 1 and 3 (9th Edition); ASM; Ohio; 1980 (Vol. 1 pp 145 151, 639 - 663, Vol. 3 pp 242 - 243)

APPENDICES

A.1 Coursework passed

To comply with the requirements for the degree of Master in Engineering, approved cousework of no less than twenty credits has to be passed. The courses attended are briefly described below.

1. AM 362 - APPLIED FUNCTIONAL ANALYSIS

The basic ideas of functional analysis are covered, with emphasis on its application to partial differential equations. Introduction to Hilbert spaces and linear operator theory. Introduction to the modern theory of partial differential equations: variational formulations of boundary-value problems: the Galerkin method and the finite element method.

(3 credits)

2. AM 363 - NUMERICAL ANALYSIS

Introduction to the theory and practise of numerical approximation. Interpolation and approximation. Numerical integration. Numerical solution of non-linear equations and of initial value problems. Accuracy of numerical solutions.

(3 credits)

3. AM 366 - TENSOR METHODS

Introduction to the methods of tensor analysis appropriate for applications in General Relativity and Physics. Tensor methods: Manifolds, co-ordinates. Tensors, tensor algebra, tensor symmetries, Canonical forms. Differential operators, covariant derivatives, Christoffel relations. Variation calculus, geodesics. Normal coordinates. Examples from various fields of applications. Curvatures, Bianchi identities, spaces of constant curvature. Special relativity.

(3 credits)

4. AM 367 - CONTINUUM MECHANICS

Introduction to tensors with applications to fluid mechanics and elasticity. Tensor algebra and analysis, basic continuum mechanics, fluid and solid mechanics, Navier-Stokes equations, the partial differential equations of elasticity.

(3 credits)

5. CE 5B8 - PLATES AND SHELLS

An introduction to the elastic theory of plates and shells. Differential equations of equilibrium. Variational methods in mechanics leading to the Ritz procedure and an introduction to the finite element method. Finite elements for plates and shells.

(2 credits)

6. CE 5B16 - INELASTIC MATERIAL BEHAVIOUR

Mathematical theories of plasticity, creep and viscoplasticity. Constitutive modelling based on thermodynamic principles and internal variables. Variational principles and theorems for limit analysis and incremental plasticity. Shakedown and bounding theorems in plasticity. Implementation of nonlinear constitutive models in finite element analysis.

(4 credits)

7. CE 5B17 - FINITE ELEMENT ANALYSIS

Generalised displacement method of analysis. Elastic energy theorems leading to basic procedures of the finite element method. Approximation and interpolation of functions. Isoparametric formulation of elements. 2-D and 3-D elements of structural mechanics. Equation solving in the computer and the structure of the finite element programme. Introduction to finite element packages for practical applications. Some advanced topics in finite element analysis.

(4 credits)

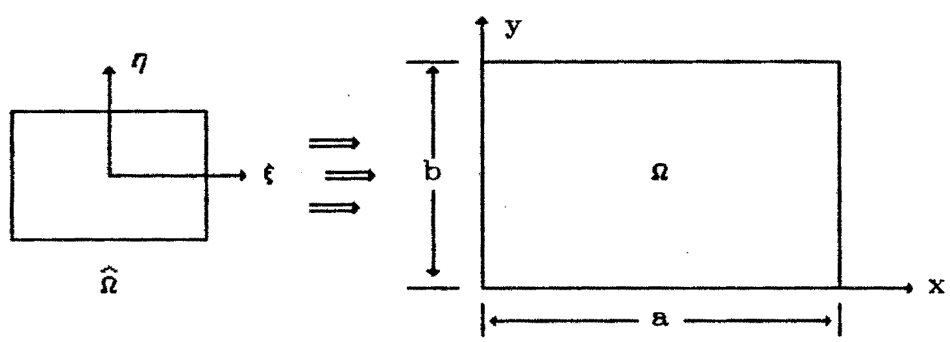
Not for credit: An introduction to the Boundary element method, and its application (Prof. G. Maier, Politecnico di Milano)

Basic introduction. Fundamental solutions: Kelvin solution. Somigliana equation. Equations derived for potential problems. Comparison between finite element method and boundary element method. Case studies. Extension of equations for elastic analysis. Overview of plasticity. Equations derived for plastic analysis. Discussion of difficulties.

(no credit)

A.2 Element matrices

The coefficient matrices are presented for a rectangular element of dimensions a wide and b heigh. This is used as a check on the routines that assemble them. Furthermore, the eigenvalues of the stiffness (conduction) matrix are found, as these have bearing on the stability of the solution. Details of this are presented in chapter 5. The information that follows refers to the element described in figure A2.1.



For this mapping, $|J| = ab/4$, which implies that

$$d\Omega = ab/4 \; d\hat{\Omega}$$

Figure A2.1 Example of rectangular isoparametric element.

General information.

Node	N_i	$\partial N_i/\partial\xi$	$\partial N_i/\partial\eta$
1	$1/4(1 - \xi)(1 - \eta)$	$-1/4(1 - \eta)$	$-1/4(1 - \xi)$
2	$1/4(1 + \xi)(1 - \eta)$	$1/4(1 - \eta)$	$-1/4(1 + \xi)$
3	$1/4(1 + \xi)(1 + \eta)$	$1/4(1 + \eta)$	$1/4(1 + \xi)$
4	$1/4(1 - \xi)(1 + \eta)$	$-1/4(1 + \eta)$	$1/4(1 - \xi)$

Table A2.1: Shape functions for 4-noded elements.

Stiffness matrix.

The stiffness matrix due to conduction is described by

$$
\begin{aligned}
[K_c] &= \int_{\hat{\Omega}} t[B]^T[k][B]\,|J|\,d\xi d\eta. \\
&= \int_{\hat{\Omega}} t\{k_x \hat{B}_i^{\xi}\hat{B}_j^{\xi} + k_y \hat{B}_i^{\eta}\hat{B}_j^{\eta}\}\,|J|\,d\xi d\eta \qquad [A2.1]
\end{aligned}
$$

Evaluating the terms individually yields:

$$
[K_c] = \frac{t|J|}{16}\begin{bmatrix}
\frac{16}{3}k_x+\frac{16}{3}k_y & -\frac{16}{3}k_x+\frac{8}{3}k_y & -\frac{8}{3}k_x-\frac{8}{3}k_y & \frac{8}{3}k_x-\frac{16}{3}k_y \\
 & \frac{16}{3}k_x+\frac{16}{3}k_y & -\frac{16}{3}k_x+\frac{8}{3}k_y & -\frac{8}{3}k_x-\frac{8}{3}k_y \\
 & & \frac{16}{3}k_x+\frac{16}{3}k_y & -\frac{16}{3}k_x+\frac{8}{3}k_y \\
 & \text{Symmetrical} & & \frac{16}{3}k_x+\frac{16}{3}k_y
\end{bmatrix}
$$

$$
= \frac{tkab}{24}\begin{bmatrix}
4 & -1 & -2 & -1 \\
-1 & 4 & -1 & -2 \\
-2 & -1 & 4 & -1 \\
-1 & -2 & -1 & 4
\end{bmatrix}. \qquad [A2.2]
$$

Damping matrix

The specific heat matrix is described by

$$[C] = \int_{\hat{\Omega}} t\rho C_p [N]^T [N] \,|J|\, d\xi d\eta .$$
$$= \int_{\hat{\Omega}} t\rho C_p \{\hat{N}_i \hat{N}_j\} \,|J|\, d\xi d\eta \qquad [A2.3]$$

Evaluating the terms individually yields:

$$[C] = \frac{t\rho C_p}{16} |J| \begin{bmatrix} (1-\xi)^2(1-\eta)^2 & (1-\xi^2)(1-\eta)^2 & (1-\xi^2)(1-\eta^2) & (1-\xi)^2(1-\eta^2) \\ & (1+\xi)^2(1-\eta)^2 & (1+\xi)^2(1-\eta^2) & (1-\xi^2)(1-\eta^2) \\ & & (1+\xi)^2(1+\eta)^2 & (1-\xi^2)(1+\eta)^2 \\ & \text{Symmetrical} & & (1-\xi)^2(1+\eta)^2 \end{bmatrix}$$

$$= \frac{tab\rho C_p}{36} \begin{bmatrix} 4 & 2 & 1 & 2 \\ 2 & 4 & 2 & 1 \\ 1 & 2 & 4 & 2 \\ 2 & 1 & 2 & 4 \end{bmatrix} . \qquad [A2.4]$$

Spectral quantities

In Cook [23] it is shown that the stability limit of a transient scheme as we have used is (for linear problems, ie where neither [K] nor [C] depend on Θ):

$$\Delta\tau_{cr} = \frac{2}{(1 - 2\alpha)\lambda_{max}}, \qquad [A2.5]$$

where λ_{max} is the largest eigenvalue of the system

$$([K] - \lambda[C])\{\theta\} = 0. \qquad [A2.6]$$

For the above arrangement, the characteristic equation reduces to

$$\lambda(\lambda - 2)(\lambda - 2)(\lambda - 4) = 0, \qquad [A2.7]$$

and hence the four eigenvalues are $\lambda_1 = 0$, $\lambda_2 = 2$, $\lambda_3 = 2$, and $\lambda_4 = 4$.

For this element, we impose a 'safe' limit on the time step size of $\Delta\tau = \frac{2}{(1 - 2\alpha)4} = \frac{1}{(2 - 4\alpha)}$ seconds. Note that as long as the element is rectangular, the element dimensions have no bearing on the recommended time step length. The values of the material properties do have a significant influence, however. The stability limiting time step length is plotted in figure A2.2 for a range of α.

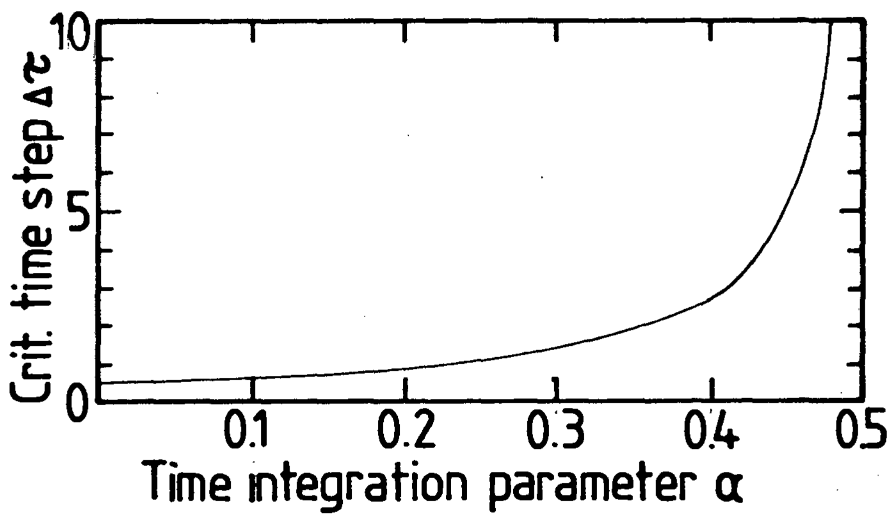


Figure A2.2: Plot of $\Delta\tau$ versus α for a rectangular element.