COMPUTER METHODS FOR THE HEAT CONDUCTION EQUATION

by

PAUL TURNER CAMERON

A thesis presented at the

University of Aston in Birmingham

for the degree of

DOCTOR OF PHILOSOPHY.

November, 1967

Summary

The merits of various numerical methods for the solution of the one and two dimensional heat conduction equation with a radiation boundary condition have been examined from a practical standpoint in order to determine accuracies and efficiencies. It is found that the use of five increments to approximate the space derivatives gives sufficiently accurate results provided the time step is not too large; further, the implicit backward difference method of Liebmann (27) is found to be the most accurate method. On this basis, a new implicit method is proposed for the solution of the three-dimensional heat conduction equation with radiation boundary conditions.

The accuracies of the integral and analogue computer methods are also investigated.

Part of this thesis has been published as American Society of Mechanical Engineers paper 67-WA/HT-17 which was written for presentation at the Winter Annual Meeting held in Pittsburgh, U.S.A., from November 12th-17th, 1967.

ACKNOWLEDGEMENTS

The Author gratefully acknowledges the assistance of the following:

Professor G. V. Jeffreys

For his encouragement and permission to carry out work in the Department.

Dr. B. Gay

For his supervision of the project and constructive comments on the preparation of the thesis.

Science Research Council

For their research studentship grant.

CONTENTS

SECTION 1

INTRODUCTION

A. Introduction

The solution of transient heat conduction problems has attracted much attention in recent years. The differential equation which represents heat conduction in a body has no general solution but is dependent on the nature of the initial and boundary conditions. Very often, no analytical solution exists, consequently approximate methods have to be used.

It appears that Richardson (33) and Hartree and Womersley (22) were the first investigators to exploit finite difference techniques for solving partial differential equations. Dusinberre (12) was first to present a practical method for the solution of heat conduction problems.

Very often, the use of difference techniques leads to quantities of calculation which are far beyond the scope of a desk calculating machine. Since the early 1950's the advancement in modern digital computer technology has been rapid and the solution of quite complex heat conduction problems by finite difference techniques has become feasible.

The finite difference technique presented by Dusinberre is simple to use but has the disadvantage of becoming unstable if too coarse a lattice density is used. Attempts to remove this drawback lead to the use of implicit methods; the first practical approach to implicit methods was that of

Crank and Nicolson (7). The Crank-Nicolson method does not have the same disadvantage as the Dusinberre method but instead it is necessary to solve a system of simultaneous equations to calculate the temperature at a particular time. It was hoped that the disadvantage of having to solve sets of simultaneous equations to calculate a set of temperatures would be outweighed by the ability to use a smaller lattice density.

Liebmann (27) presented a further implicit method which appeared to require less calculation than the method given by Crank and Nicolson.

The solution of heat conduction problems is dependent on the boundary conditions. For example, one may be considering the determination of the temperature-time history of a plate which is subject to a step change in temperature at one boundary. The application of an implicit finite difference technique to this case results in a set of linear simultaneous equations which are relatively simple to solve. Due to the recent interest in space exploration, heat transfer by radiation is becoming more important. If the plate were now considered with a radiation boundary condition, the application of an implicit technique would lead to a set of non-linear simultaneous equations which would have to be solved by an iterative method thus involving more calculation than the forementioned linear case. It may,

therefore, be more advantageous to use Dusinberre's explicit method. Fairall et al (16) used Dusinberre's method to calculate the temperature-time history of a plate subject to thermal radiation at one face. They present their results in graphical form.

Both the disadvantage of instability in Dusinberre's method and the solution of sets of simultaneous equations when implicit methods are used appears to have been removed by the method of Dufort and Frankel (11) who illustrate a stable explicit method; it appears, however, that this method has gained little support.

When the use of finite difference techniques is being considered, it is always a problem to decide the fineness of the lattice density that is required without the introduction of intolerable errors. It still appears that the accepted method of determining the necessary lattice density is trial and error. Kardas (24), however, demonstrates a method for calculating the error when a finite difference approximation is used for the case of a plate with convection at one face and the other face perfectly insulated. His method of error determination is dependent on a knowledge of an analytical solution to the problem and is therefore not directly applicable to a case where no analytical solution exists.

The methods previously mentioned have been mainly in

the context of one-dimensional problems. When one is considering a problem which cannot be reduced to the one-dimensional case, the problem is more involved, consequently problems in two and particularly three space dimensions are likely to be more time consuming and require large amounts of digital computer storage. Douglas (9) presents methods of solving heat conduction problems in two space dimensions; three of the methods are analogous to the Dusinberre, backward difference and Crank-Nicolson methods mentioned in connection with one-dimensional heat conduction problems. The fourth method, the alternating direction implicit method, has no analogue in the one-dimensional case. The alternating direction implicit method is shown to be unconditionally stable in the two-dimensional case. Allada and Quon (1) have found it to be unstable in the three-dimensional case if too large a time step is used.

The solution of heat conduction problems in three space variables is considered by Douglas and Rachford (10).

Once the set of equations is obtained by finite differencing of the original differential equation, it is necessary to consider the best methods of solving them. In the case of the Dusinberre explicit method this presents no problem but with the implicit methods it is necessary to solve the set of simultaneous equations which are formed. In the case of one-dimensional problems this is not difficult

because the set of equations form a tri-diagonal matrix which can be solved by simple elimination if the equations are linear. When the equations are non-linear an iterative technique must be applied. When a two-dimensional system is considered, the allied matrix of coefficients is still diagonal but has five non-zero terms in each equation; this system is more difficult to solve than the tri-diagonal system.

Various methods for solving such a system of equations are reviewed by Peaceman and Rachford (31) who make estimates of the calculations requirements for each method. Frankel (18) reviews iterative methods for solving the Laplace (two dimensional steady state heat conduction) equation, which is similar to the two-dimensional transient equation. A full range of iterative methods for solving sets of simultaneous equations is discussed by Varga (40). Liebmann (27) discusses relaxation techniques applied to the backward difference method.

The difference methods mentioned require finite differencing of both the space and time derivatives; if instead, only the space derivative were differenced, one would obtain a set of ordinary differential equations which are suitable for an analogue computer solution. Zerkle and Sunderland (42) present charts of solutions to one-dimensional heat conduction problems using an analogue

computer to simulate a plate which is subject to radiation heat transfer. Approximate analytical solutions were used to estimate the accuracy of the analogue computer.

An analogue computer can be an effective method of solving heat conduction problems but has the disadvantage that large amounts of equipment are required for all but the simplest problems. The use of a hybrid computer may have possibilities in the solution of more involved heat conduction problems where it is possible to time-share items of analogue equipment to solve a set of equations; a task which the analogue computer does well. The solution of sets of equations can be time consuming on a digital computer.

Schneider (37) presents graphs representing the transient temperature distribution in solids, several of which have been calculated using the integral method. This method is analogous to boundary layer theory and is described by Goodman(20). Schneider (38) illustrates the integral method applied to a plate with radiation and adiabatic boundary conditions. Roberts (36) applied the integral method to the calculation of temperature distribution in cylinders heated by radiation. Reynolds and Dolton (35) present the integral method applied to various shaped bodies. Siddall (39) also discusses the use of the integral method. Gay (19) compares the integral method for a plate subject to radiation and adiabatic boundary conditions with that obtained using an

analogue computer and the Dusinberre and Crank Nicolson difference methods.

The use of a general digital computer program to solve transient heat conduction problems does not appear to be popular; this is understandable considering the problems it would present. A search through the literature revealed the existence of one such program which is discussed by Campbell and Vollenweider (5). It appeared from the discussion that this program could solve quite complex heat conduction problems but required a large amount of input data. It seemed capable of taking account of any shape, and allowance is made for all modes of heat transfer at boundaries. The body under consideration is broken down into a number of finite cells, input data being required for each cell. Wagner (41) discusses a computer program for the IBM 650 computer for solving one dimensional transient and steady state heat conduction problems. Fox et al (17) discuss the use of numerical approximations in heat transfer as applied to an IBM 7090 computer.

From the foregoing, it is apparent that there are numerous methods of solving transient heat conduction problems, each method having its own advantages and disadvantages. An effort is made in this thesis to evaluate the practicability of various methods and present criteria for their use.

B. Mathematical Models

The three-dimensional heat conduction equation of Fourier can be written:

$$C\rho \frac{\partial T}{\partial t} = \frac{\partial}{\partial x}\left(k_x \frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(k_y \frac{\partial T}{\partial y}\right) + \frac{\partial}{\partial z}\left(k_z \frac{\partial T}{\partial z}\right) \qquad (1.1)$$

which represents the temperature-time history of a point in a three-dimensional body.

Equation (1.1) has no specific solution but requires the knowledge of certain initial and boundary conditions to define the problem completely. For example, if a plate, initially (t=0) at a uniform temperature throughout, insulated at one face and radiating to a zero sink temperature at the other face (when t>0) were considered, equation (1.1) would reduce to:

$$C\rho \frac{\partial T}{\partial t} = \frac{\partial}{\partial x}\left(k_x \frac{\partial T}{\partial x}\right) \qquad (1.2)$$

with the initial and boundary conditions:

$$t < 0 \qquad T(x,0) = T_i \qquad (1.3)$$

$$t \geq 0 \qquad -k\frac{\partial T}{\partial x}(0,t) = \sigma e T^4(0,t) \qquad (1.4)$$

$$\frac{\partial T}{\partial x}(L,t) = 0 \qquad (1.5)$$

If it is further assumed that the properties of the plate are independent of temperature, equation (1.2) reduces to:

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2} \qquad (1.6)$$

with the same initial and boundary conditions as above.

Equation (1.6) can be expressed in dimensionless form:

$$\frac{\partial \theta}{\partial \tau} = \frac{\partial^2 \theta}{\partial X^2} \tag{1.7}$$

with the corresponding initial and boundary conditions:

$$\tau < 0 \qquad \theta(X,0) = 1 \tag{1.8}$$

$$\tau \geq 0 \qquad -\frac{\partial \theta}{\partial X}(0,\tau) = R\theta^4(0,\tau) \tag{1.9}$$

$$\frac{\partial \theta}{\partial X}(1,\tau) = 0 \tag{1.10}$$

The solution of equation (1.7) with the initial and boundary conditions expressed as equations (1.8) to (1.10) has attracted most attention in this thesis.

Making equation (1.6) dimensionless greatly simplifies the analysis as the solutions obtained for equation (1.7) can be simply applied to a problem for a plate of any thickness and any initial temperature provided that temperature is uniform throughout the plate.

C. Methods of solving one-dimensional heat conduction equations.

(i) Explicit

The first practical approach to transient heat conduction problems was that of Dusinberre (12), whose use of the explicit method has attracted much attention.

Dusinberre analysed the problem of transient conduction in one dimension through an insulated bar of material by making heat balances over finite divisions of the bar. A similar case to an insulated bar is the plate which is effectively a one-dimensional heat conductor. Fig. 1.1 illustrates the plate divided into five nodal sections (capacities).

Making a heat balance on node 2 (say) over a finite period of time (Δt):

input - output = accumulation

$$k\Delta t\left(T_{(1,t)}-T_{(2,t)}\right) - k\Delta t\left(T_{(2,t)}-T_{(3,t)}\right) = \Delta x^2 \rho C\left(T_{(2,t+\Delta t)}-T_{(2,t)}\right) \quad (1.11)$$

$$\therefore T_{(2,t+\Delta t)} = T_{(2,t)} + \frac{k\Delta t}{\Delta x^2 \rho C}\left(T_{(1,t)} - 2T_{(2,t)} + T_{(3,t)}\right) \quad (1.12)$$

which in dimensionless form is:

$$\theta_{(2,\tau+\Delta\tau)} = \theta_{(2,\tau)} + \frac{\Delta\tau}{\Delta X^2}\left(\theta_{(1,\tau)} - 2\theta_{(2,\tau)} + \theta_{(3,\tau)}\right) \quad (1.13)$$

Equations (1.12) and (1.13) are the basis of the Dusinberre explicit method.

Equation (1.13) can be derived in a more mathematical manner using the principle of finite differences. If the

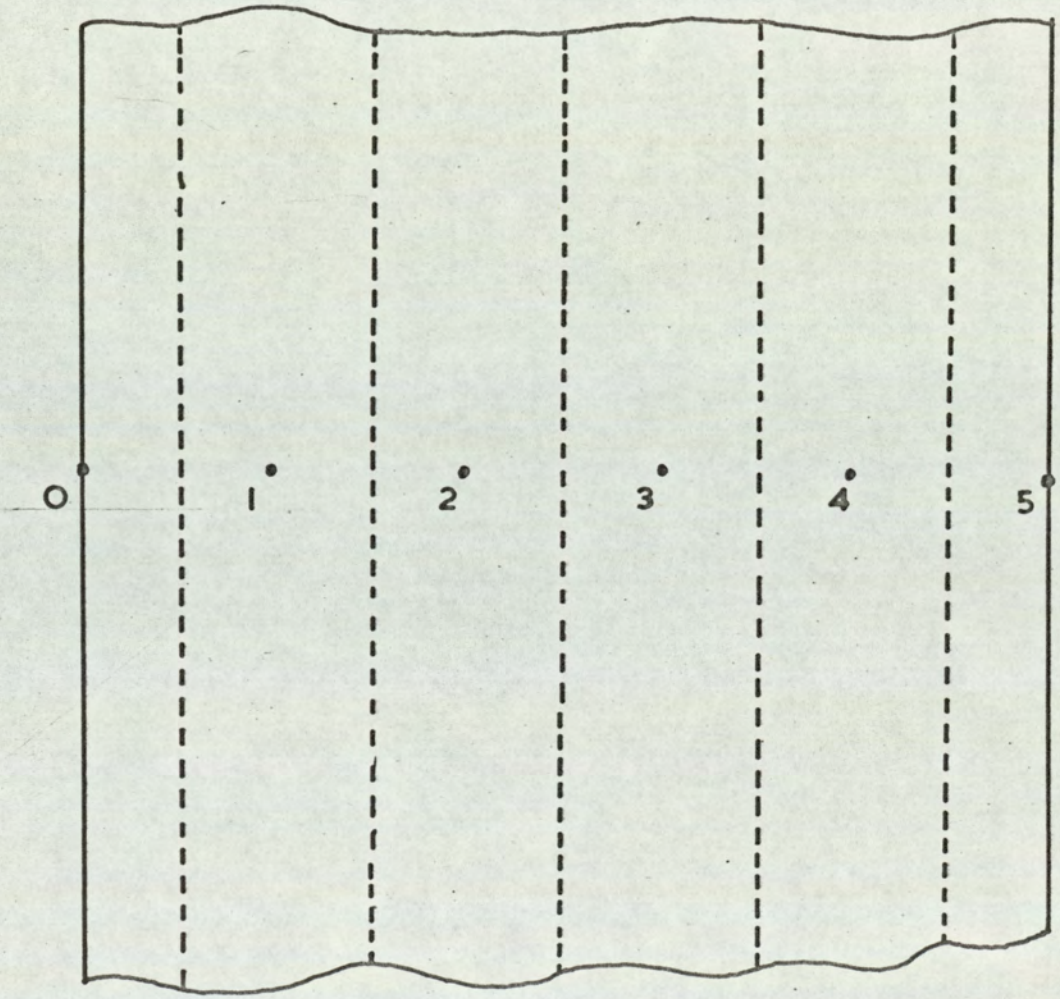


FIG 1.1 CROSS SECTION OF AN INFINITE PLATE DIVIDED INTO FIVE NODES.

plate is divided into segments of equal thickness across its width as shown in fig. 1.2, by Taylor's theorem:-

$$\theta_{(m+1,\tau)} = \theta_{(m,\tau)} + \Delta X\left(\frac{\partial\theta}{\partial X}\right)_{(m,\tau)} + \frac{\Delta X^2}{2!}\left(\frac{\partial^2\theta}{\partial X^2}\right)_{(m,\tau)} + \frac{\Delta X^3}{3!}\left(\frac{\partial^3\theta}{\partial X^3}\right)_{(m,\tau)} + \cdots \qquad (1.14)$$

and:

$$\theta_{(m-1,\tau)} = \theta_{(m,\tau)} - \Delta X\left(\frac{\partial\theta}{\partial X}\right)_{(m,\tau)} + \frac{\Delta X^2}{2!}\left(\frac{\partial^2\theta}{\partial X^2}\right)_{(m,\tau)} - \frac{\Delta X^3}{3!}\left(\frac{\partial^3\theta}{\partial X^3}\right)_{(m,\tau)} + \cdots \qquad (1.15)$$

adding equations (1.14) and (1.15), then re-arranging:

$$\left(\frac{\partial^2\theta}{\partial X^2}\right)_{(m,\tau)} = \frac{\theta_{(m+1,\tau)} - 2\theta_{(m,\tau)} + \theta_{(m-1,\tau)}}{\Delta X^2} + O(\Delta X^2) \qquad (1.16)$$

where $O(\Delta X^2)$ denotes terms containing ΔX^2 and higher.

A similar analysis in the τ direction leads to:

$$\left(\frac{\partial\theta}{\partial\tau}\right)_{(m,\tau)} = \frac{\theta_{(m,\tau+\Delta\tau)} - \theta_{(m,\tau)}}{\Delta\tau} + O(\Delta\tau) \qquad (1.17)$$

Equating equations (1.16) and (1.17) when higher orders are ignored leads to:

$$\theta_{(m,\tau+\Delta\tau)} = \theta_{(m,\tau)} + \frac{\Delta\tau}{\Delta X^2}\left(\theta_{(m+1,\tau)} - 2\theta_{(m,\tau)} + \theta_{(m-1,\tau)}\right) \qquad (1.18)$$

which is the same as equation (1.13).

Thus the temperature in the plate is known only at the nodal points in the plate. Knowing the temperatures at time τ, the nodal temperatures at $\tau+\Delta\tau$ can be calculated by the use of equation (1.18).

Equation (1.18) has the drawback of instability when $\Delta\tau$ is too large.

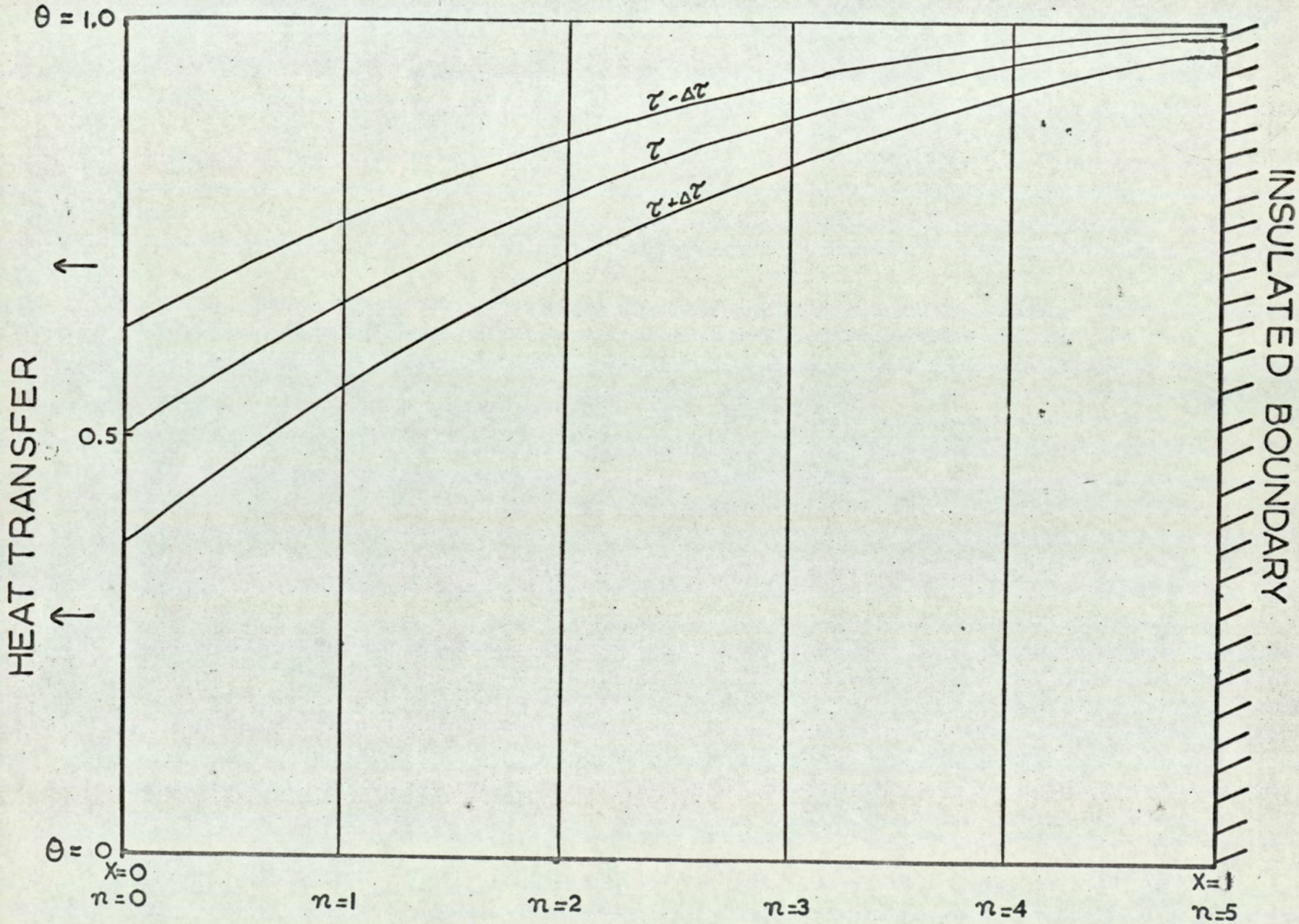


FIG 1.2 TEMPERATURE-TIME HISTORY

(ii) Backward difference

To overcome the drawback of instability in the Dusinberre explicit method, Liebmann (27) proposed evaluating the difference analogue for $\partial^2\theta/\partial X^2$ at $\tau+\Delta\tau$ rather than at τ. This leads to:

$$\frac{\theta_{(n,\tau+\Delta\tau)} - \theta_{(n,\tau)}}{\Delta\tau} = \frac{\theta_{(n+1,\tau+\Delta\tau)} - 2\theta_{(n,\tau+\Delta\tau)} + \theta_{(n-1,\tau+\Delta\tau)}}{\Delta X^2} \qquad (1.19)$$

when the higher orders of ΔX and $\Delta\tau$ are ignored. Re-arranging equation (1.19):

$$-M\theta_{(n+1,\tau+\Delta\tau)} + (1+2M)\theta_{(n,\tau+\Delta\tau)} - M\theta_{(n-1,\tau+\Delta\tau)} = \theta_{(n,\tau)} \qquad (1.20)$$

Each equation of type (1.20), for each nodal point, contains three unknowns but the whole set form simultaneous equations with either N or N + 1 unknowns, depending on the type of boundary conditions which are being considered; the step change boundary condition requires the solution of N and derivative boundary condition of N + 1 equations. The backward difference method is sometimes referred to as the "implicit method". Knowing the plate temperatures at time, τ, those at $\tau+\Delta\tau$ are evaluated by the solution of the simultaneous equations. Thus it is evident that for each time level, the price of stability is the solution of simultaneous equations.

(iii) Crank-Nicolson

Crank and Nicolson (7) suggest the calculation of the difference analogue for $\partial^2\theta/\partial x^2$ be taken at $\tau+\frac{1}{2}\Delta\tau$ Thus, averaging the analogues used in the explicit and implicit methods leads to:

$$\frac{\partial^2\theta}{\partial x^2} \simeq \frac{1}{2}\left\{\frac{\theta_{(m+1,\tau)}-2\theta_{(m,\tau)}+\theta_{(m-1,\tau)}}{\Delta x^2} + \frac{\theta_{(m+1,\tau+\Delta\tau)}-2\theta_{(m,\tau+\Delta\tau)}+\theta_{(m-1,\tau+\Delta\tau)}}{\Delta x^2}\right\} \quad (1.21)$$

Equating this difference analogue to that for $\partial\theta/\partial\tau$ leads to the final equation:

$$[\theta_{(m+1,\tau+\Delta\tau)}-2(1+s)\theta_{(m,\tau+\Delta\tau)}+\theta_{(m-1,\tau+\Delta\tau)}] = -[\theta_{(m+1,\tau)}-2(1-s)\theta_{(m,\tau)}+\theta_{(m-1,\tau)}] \quad (1.22)$$

Equation (1.22) is similar to equation (1.20) except that more calculation is required to evaluate the right hand side of equation (1.22). It would be expected that the Crank-Nicolson formulation should be more accurate than either the implicit or explicit methods as the difference analogue for $\partial^2\theta/\partial x^2$ is taken at $\tau+\frac{1}{2}\Delta\tau$ which is the same point as that at which the analogue for $\partial\theta/\partial\tau$ is taken.

(iv) Summary equation

Equations (1.18), (1.19) and (1.20) can be summarised by the equation:

$$\frac{\theta_{(m,\tau+\Delta\tau)}-\theta_{(m,\tau)}}{\Delta\tau} = (1-K)[\theta_{(m+1,\tau)}-2\theta_{(m,\tau)}+\theta_{(m-1,\tau)}]/\Delta x^2 + K[\theta_{(m+1,\tau+\Delta\tau)}-2\theta_{(m,\tau+\Delta\tau)}+\theta_{(m-1,\tau+\Delta\tau)}]/\Delta x^2 \quad (1.23)$$

when K = 0, the Dusinberre explicit method is obtained;

$K = 1$ corresponds to the implicit method; and $K = \frac{1}{2}$ to the Crank-Nicolson method.

It is not always necessary to use values of $K = 0, \frac{1}{2}$ or 1 since it may be found that values of K intermediate to these values may give accurate results.

D. Comments on the use of the difference methods

There has been much discussion in the literature on the various merits of the difference methods.

It is obvious that the advantage of the implicit method lies in the ability to use larger time steps in the solution of a problem thus possibly saving computer time.

Dusinberre (13) defends the use of the explicit method with the statement that large time steps usually result in large truncation errors, hence the solution becomes inaccurate. To support his statements he cites the problem of a cylinder, initially at 1000°C and having its surface temperature suddenly reduced to 0°C. The results (shown in table 1.1) obtained using the explicit method with N = 4 and M = 0.4 for this one dimensional problem are compared with the analytical solution for the centre line temperature. The maximum error obtained was only 1.2%. The maximum error is reached (at the fifth time step) and gradually decreases thereafter until at the thirteenth time step it reaches 0.3%. Dusinberre states that his explicit formulation required 3N multiplications and N additions for each time step. He compares the results obtained using the explicit method with those using the implicit method with a time step five times as great and finds that the maximum discrepancy is 7.8% at the centre line but at the outer surface it is 10.8%. (A discontinuity exists at the outer surface). Thus it is concluded that the

TABLE 1.1

DUSINBERRE'S RESULTS

$\frac{t}{\Delta t}$	EXPLICIT	ANALYTICAL	% ERROR	IMPLICIT	% DISCREPANCY
0	1000	1000	0		
1	1000	1000	0		
2	1000	1000	0		
3	1000	997	0.3		
4	993	987	0.6		
5	977	965	1.2	899	- 7.8
6	944	933	1.1		
7	904	893	1.1		
8	857	848	0.9		
9	808	801	0.7		
10	760	754	0.6	743	- 1.7
11	712	707	0.5		
12	665	662	0.3		
13	621	618	0.3		
14	NOT REPORTED	-	-		
15	539	-	-	586	+ 4.7

ability to use larger time steps is not a sufficient enough reason to recommend the implicit method.

Dusinberre only cited one problem; that where there is a step change in the boundary temperature which is a severe condition. An analysis made where the boundary temperature is varying much less rapidly may show the implicit method to be superior. Liebmann (27) used a graded set of time steps, small where the temperatures varied rapidly and large where they varied more slowly.

Anderson et al (2) support the implicit method by citing Liebmanns example where he covered the time range in 36 time steps with a maximum error of 2%. The same problem using the explicit method would have taken 10,000 time steps. They also cite an example (3) where to use the explicit method would have required the use of an IBM 704 computer for 3 hours but using the implicit method, the problem was completed in 12 minutes. Anderson et al conclude that the implicit is as accurate as the explicit method but the ability to use larger time steps, where this is possible due to a small truncation error, is advantageous.

Greenwood (21) supports the use of Liebmann's method in preference to that of Crank and Nicolson because of its superior stability, although the Crank-Nicolson method has the advantage of accuracy. In his analysis, Greenwood illustrates the example of a plate with the initial temperature

distribution:

$$T = 100\left(1 - \frac{x}{L}\right) - 15 \sin\left(\frac{2\pi x}{L}\right)$$

and fixed boundary temperatures.

He compares the results obtained using the Crank-Nicolson and implicit methods with an analytical solution. The plate is divided into 4 space increments and M = 4 (a very high value) is used. The results are compared at all the nodal points. It is clearly seen (Table 1.2) that the implicit method gives accurate results but the Crank-Nicolson method is inaccurate. For the first time step there is a positive error but at the second the error is negative. Greenwood supports the use of high M by stating that this may be unavoidable when the thermal diffusivity is very temperature dependent. He mentioned a problem where the thermal diffusivity varied with temperature between 1 and 1000. By a quick calculation he demonstrates the oscillatory nature of the Crank-Nicolson, as compared with the Liebmann method by showing that when $M \geq 100$ (the instantaneous solution is then $\partial^2 T/\partial x^2 = 0$) the correct result for $\partial^2 T/\partial x^2$ is obtained using Liebmanns method whereas when the Crank-Nicolson method is used the result $-\partial^2 T/\partial x^2$ is returned thus showing the oscillatory nature of the Crank-Nicolson method.

TABLE 1.2

GREENWOOD'S RESULTS.

x/L =		0	0.25	0.50	0.75	1.0
time, t		TEMPERATURES				
0		100	60	50	40	0
$L^2/4\alpha$	A	100	74.9992	50	25.0008	0
	B	100	84	50	16	0
	C	100	73.333	50	26.667	0

A Analytical Solution

B Crank-Nicolson

C Liebmann

E. Boundary Conditions

In this thesis, the main concern is the evaluation of temperature-time histories of bodies with non-linear boundary conditions. The non-linear case of transient heat conduction with a radiation boundary condition has no analytical solution unlike the linear cases of step change and convection. These cases have been considered in order to make comparisons between the numerical and analytical solutions. It is to be expected that a numerical solution which gives accurate results in the convection and step change boundary cases will also give an accurate result in the case of a radiation boundary condition; the step change is equivalent to R = ∞ with the proviso that the boundary temperature is defined exactly in the step change case.

The three boundary conditions are expressed mathematically as follows:

(a) Step Change

$$\tau \leq 0 \quad 0 \leq X \leq 1 \quad \theta_{(m,\tau)} = 1 \tag{1.24}$$

$$\tau > 0 \quad X = 0 \quad \theta_{(0,\tau)} = 0 \tag{1.25}$$

(b) Convection

$$\tau \leq 0 \quad 0 \leq X \leq 1 \quad \theta_{(m,\tau)} = 1 \tag{1.26}$$

$$\tau > 0 \quad X = 0 \quad \left(\frac{\partial \theta}{\partial X}\right)_{(0,\tau)} = Bi\,\theta_{(0,\tau)} \tag{1.27}$$

(c) Radiation

$$\tau \leq 0 \qquad 0 \leq X \leq 1 \qquad \theta_{(n,0)} = 1 \tag{1.28}$$

$$\tau > 0 \qquad X = 0 \qquad \left(\frac{\partial \theta}{\partial X}\right)_{(0,\tau)} = R\theta^{4}_{(0,\tau)} \tag{1.29}$$

The adiabatic boundary condition at the other face is expressed:

$$\tau > 0 \qquad X = 1 \qquad \left(\frac{\partial \theta}{\partial X}\right)_{(1,\tau)} = 0 \tag{1.30}$$

F. Finite difference approximations at the boundaries.

In the case of a step change boundary condition, the boundary temperature is defined at the surface where heat transfer is occuring. However, for the first time step it is still necessary to approximate the boundary temperature. Examination of equation (1.18) shows that if the temperature $\theta_{(1,\Delta\tau)}$ is required, the substitution of $\theta_{(0,0)}=1$ would return a value of $\theta_{(1,\Delta\tau)}=1$ which indicates that no progress would be made with the calculation.

From equations (1.24) and (1.25) $\theta_{(0,0)}=1$ and $\theta_{(0,\Delta\tau)}=0$ therefore a reasonable approximation to substitute into equation (1.18) for the first time step would be $\theta_{(0,0)}=0.5$ and thereafter $\theta_{(0,\tau)}=0$.

The step change boundary condition has been investigated by Pearson (32) and Elrod (15). Pearson used the Crank-Nicolson method and considered various approximations to $\theta_{(0,0)}$ for the first time step. He compares his results with the analytical solution to the problem and shows that for the first time step the results are inaccurate (even using the high value of N = 100) but as τ increases the results become more accurate. On this basis it is suggested that if the value of $\theta_{(n,\tau)}$ is required, $\Delta\tau$ be divided into a number of smaller time steps thus by the time $\Delta\tau$ is reached the solution will be accurate. Pearson also uses the explicit method for solution of the problem. He concludes with a formula for the

optimum value of $\theta_{(0,0)}$.

Elrod develops a method for a weighted approximation to the boundary temperature. His results appear to be very accurate, when the explicit method is used, even for the first time step.

When there is convection or radiation at the boundary X = 0, the boundary temperature is determined by the derivative with respect to X, consequently this temperature varies with time and has to be represented by a finite difference approximation.

Examination of the various finite difference equations, e.g. (1.18), shows that if $\theta_{(0,\tau)}$ is being considered, a numerical value for the fictitious temperature $\theta_{(-1,\tau)}$ must be calculated.

Using a Taylor series expansion at the boundary X = 0:

$$\theta_{(1,\tau)} = \theta_{(0,\tau)} + \Delta X\left(\frac{\partial\theta}{\partial X}\right)_{(0,\tau)} + \frac{\Delta X^2}{2!}\left(\frac{\partial^2\theta}{\partial X^2}\right)_{(0,\tau)} + \frac{\Delta X^3}{3!}\left(\frac{\partial^3\theta}{\partial X^3}\right)_{(0,\tau)} + \cdots \quad (1.30)$$

and:

$$\theta_{(-1,\tau)} = \theta_{(0,\tau)} - \Delta X\left(\frac{\partial\theta}{\partial X}\right)_{(0,\tau)} + \frac{\Delta X^2}{2!}\left(\frac{\partial^2\theta}{\partial X^2}\right)_{(0,\tau)} - \frac{\Delta X^3}{3!}\left(\frac{\partial^3\theta}{\partial X^3}\right)_{(0,\tau)} + \cdots \quad (1.31)$$

From equation (1.31) it is seen that:

$$\left(\frac{\partial\theta}{\partial X}\right)_{(0,\tau)} = \frac{\theta_{(0,\tau)} - \theta_{(-1,\tau)}}{\Delta X} \quad (1.32)$$

when terms of the first order and higher are ignored. However, subtracting equation (1.31) from (1.30) leads to:

$$\left(\frac{\partial\theta}{\partial X}\right)_{(0,\tau)} = \frac{\theta_{(1,\tau)} - \theta_{(-1,\tau)}}{2\Delta X} \quad (1.33)$$

when terms of the second order and higher are ignored.

It is to be expected, therefore, that the boundary approximation given by equation (1.33) would be more accurate than that given by equation (1.32)

The approximation given by equation (1.33) can often be used. Taking the example of the convective boundary condition, from equations (1.27) and (1.33):

$$\frac{\theta_{(1,\tau)} - \theta_{(-1,\tau)}}{2\Delta X} = Bi\,\theta_{(0,\tau)} \tag{1.34}$$

$$\therefore \theta_{(-1,\tau)} = \theta_{(1,\tau)} - 2\Delta X\, Bi\, \theta_{(0,\tau)} \tag{1.35}$$

Thus a value for $\theta_{(-1,\tau)}$ can be substituted into the difference equations for the boundary temperature.

A similar analysis for a finite difference approximation about the adiabatic boundary leads to the approximation for $\theta_{(N+1,\tau)}$:

$$\left(\frac{\partial\theta}{\partial X}\right)_{(N,\tau)} = \frac{\theta_{(N+1,\tau)} - \theta_{(N-1,\tau)}}{2\Delta X} = 0 \tag{1.36}$$

$$\therefore \theta_{(N+1,\tau)} = \theta_{(N-1,\tau)} \tag{1.37}$$

Equation (1.36) always returns a correct value for $(\partial^2\theta/\partial X^2)_{(N,\tau)}$ when substitution is made, however it is readily seen that an approximation using equation (1.32) would not be as accurate.

An interesting case of boundary approximations is in the case of two plates of different materials in perfect thermal contact at the boundaries X = 1 as shown in fig. 1.3. It has been usual practice to represent the continuity of heat flux through the plate by a first order approximation

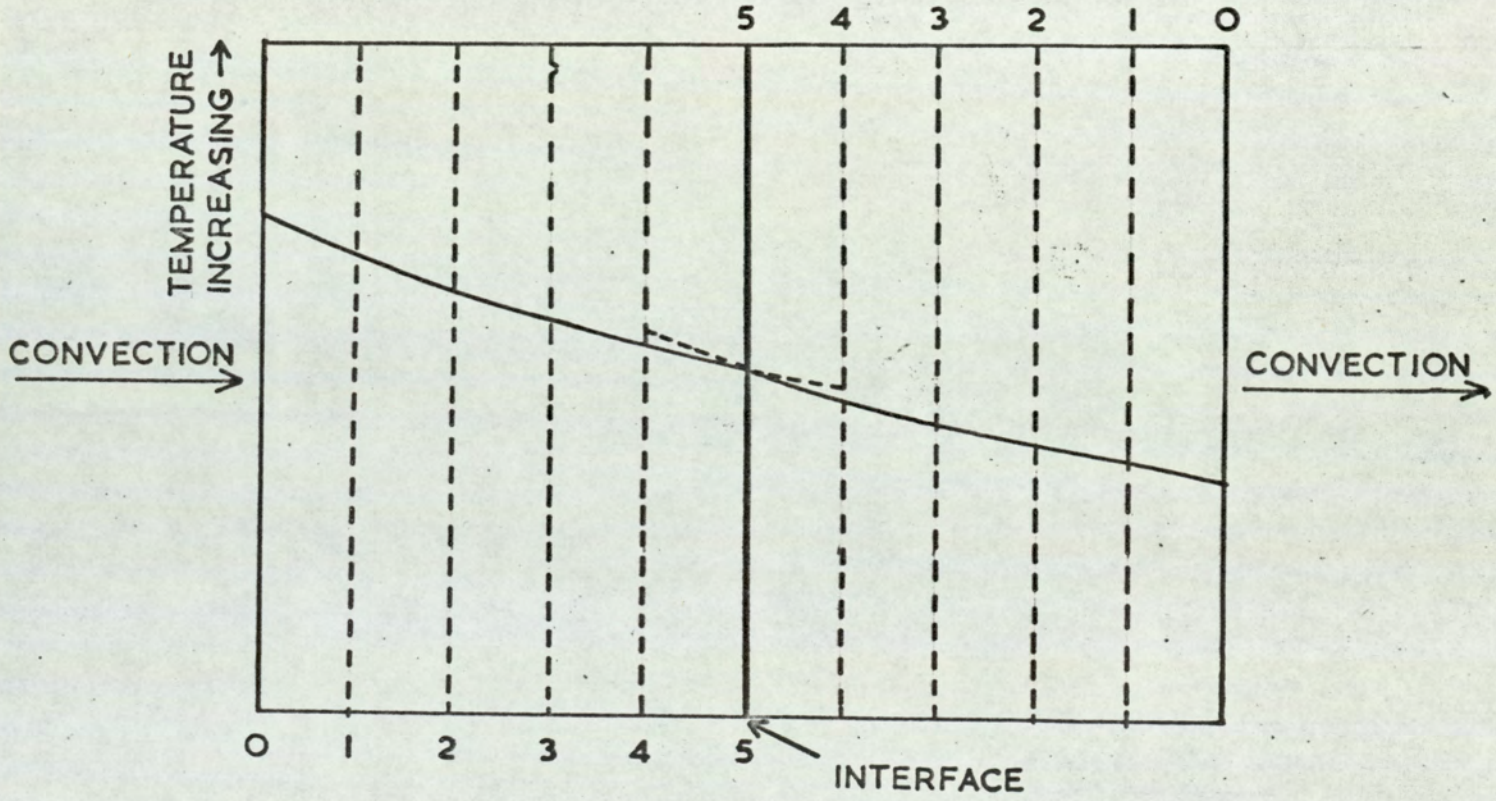


FIG 1.3 TEMPERATURE PROFILE IN A COMPOSITE PLATE

(similar to that given by equation (1.32)), but this can be improved upon:

Considering the left-hand plate, when say the explicit method is used at node 5, an approximation is required for the fictitious temperature $T_{(6,t)}$. Since the flow of heat from the left hand plate is the same as that into the right hand plate:

$$q = -k_1\left(\frac{\partial T}{\partial x}\right)_{(5,t)LS} = -k_2\left(\frac{\partial T}{\partial x}\right)_{(5,t)RS} \tag{1.38}$$

$$\therefore \frac{k_1(T_{(6,t)} - T_{(4,t)})_{LS}}{2\Delta x_1} = \frac{k_2(T_{(4,t)} - T_{(6,t)})_{RS}}{2\Delta x_2} = q \tag{1.38a}$$

where the subscripts refer to the left and right plates respectively.

Difference equation (1.12) can be written for the interface, whether it is approached from the left side or the right side as:

$$T_{(5,t+\Delta t)} = T_{(5,t)} + \frac{k_1\Delta t}{\rho_1 C_{p_1}\Delta x_1^2}\left(T_{(6,t)LS} - 2T_{(5,t)} + T_{(4,t)LS}\right) \tag{1.38c}$$

$$T_{(5,t+\Delta t)} = T_{(5,t)} + \frac{k_2\Delta t}{\rho_2 C_{p_2}\Delta x_2^2}\left(T_{(6,t)RS} - 2T_{(5,t)} + T_{(4,t)RS}\right) \tag{1.38d}$$

substitution of $T_{(6,t)}$ for both the left and right side using equation (1.38a) leads to:

$$T_{(5,t+\Delta t)} = T_{(5,t)} + M_1\left[\frac{2\Delta x_1 q}{k_1} - 2T_{(5,t)} + 2T_{(4,t)LS}\right] \tag{1.39}$$

$$T_{(5,t+\Delta t)} = T_{(5,t)} + M_2\left[-\frac{2\Delta x_2 q}{k_2} - 2T_{(5,t)} + 2T_{(4,t)RS}\right] \tag{1.39a}$$

Elimination of $T_{(5,t+\Delta t)}$ from equations (1.39) and (1.39a) leads to:

$$q = \frac{k_1 k_2}{M_1 k_2 \Delta x_1 + M_2 k_1 \Delta x_2}\left[M_1\left(T_{(5,t)} - T_{(4,t)LS}\right) - M_2\left(T_{(5,t)} - T_{(4,t)RS}\right)\right] \quad (1.40)$$

Therefore, q can be calculated from equation (1.40) and substitution for q into equation (1.39) leads to an equation for the calculation of $T_{(5,t+\Delta t)}$. The boundary approximation is second order.

A second method of approximating the interface would be to use an equation similar to (1.32) to calculate the heat flux. It is to be expected, however, that this would be less accurate. Difference equations can be very inaccurate in the region of discontinuities.

The approximate solutions by finite difference methods for the cases where there are derivative boundary conditions have been considered by Back (4), Elrod (15) and Lynn and Meyer (28)

Elrod appears to have given the most accurate method. This method is used in conjunction with the explicit method, and uses essentially a weighted mean of the boundary temperature and the internal plate temperature. The values of the weighting coefficients are dependent on the Biot number.

Back presents a method which is claimed to overcome the disadvantage of instability in the Dusinberre method. He evaluates the convective boundary temperature as the average over time interval Δt. Results are compared with Elrod's and Dusinberre's which shows Back's method to be inaccurate

at small values of time but at larger values Back's method seems to be superior to Dusinberre's. The results obtained using Elrod's method are much superior to those obtained using the other methods.

Lynn and Meyer compare Back's explicit method with the Crank-Nicolson method for both surface and internal temperatures. They conclude that the Crank-Nicolson method produces more accurate results than Back's method. They also state that the Crank-Nicolson method was used with increasing time steps; for example, in stepping from $3\Delta t$ to $6\Delta t$ in one time step resulted in an error of $< 1\%$ in the surface temperature.

Lynn and Meyer's results are reproduced in tables 1.3 and 1.4.

TABLE 1.3

RESULTS PRESENTED BY LYNN AND MEYER.

M = 0.5 Bi = 5 N = 10 X = 0

TIME	$\theta_{(0,t)}$ (EXACT)	$\theta_{(0,t)}$ (EXPLICIT)	$E_{(0,t)}$ %	$\theta_{(0,t)}$ CRANK-NICOLSON	$E_{(0,t)}$ %
0	1.000	1.000		1.000	
Δt	0.699	0.714	2.2	0.699	0.1
$2\Delta t$	0.616	0.633	2.8	0.617	0.2
$3\Delta t$	0.563	0.569	1.0	0.563	0
$4\Delta t$	0.523	0.528	1.0	0.523	-0.1
$5\Delta t$	0.493	0.495	0.4	0.492	-0.2
$6\Delta t$	0.467	0.469	0.4	0.466	-0.2

TABLE 1.4

RESULTS PRESENTED BY LYNN AND MEYER

M = 0.5 Bi = 20 N = 10 X = 0

TIME	$\theta_{(0,t)}$ (EXACT)	$\theta_{(0,t)}$ (EXPLICIT)	$E_{(0,t)}$ %	$\theta_{(0,t)}$ CRANK-NICOLSON	$E_{(0,t)}$ %
0	1.000	1.000		1.000	
Δt	0.338	0.200	-41	0.172	-49
$2\Delta t$	0.256	0.280	+ 9	0.272	+ 6
$3\Delta t$	0.214	0.192	-10	0.199	- 7
$4\Delta t$	0.188	0.189	+ 0.5	0.185	- 1.6
$5\Delta t$	0.170	0.162	- 5	0.166	- 2
$6\Delta t$	0.157	0.154	- 2	0.154	- 2

G. Finite difference approximations in two dimensions

The same differencing principles which were used in the one-dimensional case can be used in two space dimensions. The equation to be considered is:

$$\frac{\partial T}{\partial t} = \alpha \left[\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right] \tag{1.41}$$

assuming the case under consideration is a long bar of square cross section, insulated on two adjacent sides, then equation (1.36) can be made dimensionless:

$$\frac{\partial \theta}{\partial \tau} = \frac{\partial^2 \theta}{\partial X^2} + \frac{\partial^2 \theta}{\partial Y^2} \tag{1.42}$$

The initial and boundary conditions (assuming these to be radiation) are expressed:

$$\tau = 0 \quad 0 \le X \le 1 \quad 0 \le Y \le 1 \quad \theta(X, Y, 0) = 1 \tag{1.43}$$

$$\tau > 0 \quad 0 \le Y \le 1 \quad X = 0 \quad -\frac{\partial \theta}{\partial X}(0, Y, \tau) = R\theta^4(0, Y, \tau) \tag{1.44}$$

$$0 \le X \le 1 \quad Y = 0 \quad -\frac{\partial \theta}{\partial Y}(X, 0, \tau) = R\theta^4(X, 0, \tau) \tag{1.45}$$

$$0 \le Y \le 1 \quad X = 1 \quad \frac{\partial \theta}{\partial X}(1, Y, \tau) = 0 \tag{1.46}$$

$$0 \le X \le 1 \quad Y = 1 \quad \frac{\partial \theta}{\partial Y}(X, 1, \tau) = 0 \tag{1.47}$$

The following approximations to equation (1.42) have previously been used:

(i) Explicit

This method is analogous to equation (1.18):

$$\frac{\theta_{(m,k,\tau+\Delta\tau)} - \theta_{(m,k,\tau)}}{\Delta\tau} = \frac{\theta_{(m+1,k,\tau)} - 2\theta_{(m,k,\tau)} + \theta_{(m-1,k,\tau)}}{\Delta x^2}$$

$$+ \frac{\theta_{(m,k+1,\tau)} - 2\theta_{(m,k,\tau)} + \theta_{(m,k-1,\tau)}}{\Delta y^2} \qquad (1.48)$$

Equation (1.48) is arranged to enable $\theta_{(m,k,\tau+\Delta\tau)}$ to be calculated explicitly from $\theta_{(m,k,\tau)}$:

$$\theta_{(m,k,\tau+\Delta\tau)} = \theta_{(m,k,\tau)} + M\left(\theta_{(m+1,k,\tau)} + \theta_{(m-1,k,\tau)} + \theta_{(m,k+1,\tau)} + \theta_{(m,k-1,\tau)} - 4\theta_{(m,k,\tau)}\right) \qquad (1.49)$$

The use of the explicit method in the two-dimensional case is theoretically even less desirable than in the one-dimensional case because the maximum time step is only half that possible in the one dimensional case due to the more stringent stability requirement.

(ii) Backward difference

The backward difference method is the two-dimensional analogue of equation (1.20). Evaluating the space derivatives of equation (1.41) at time $\tau+\Delta\tau$ instead of at τ .

$$-M\theta_{(m+1,k,\tau+\Delta\tau)} - M\theta_{(m-1,k,\tau+\Delta\tau)} - M\theta_{(m,k+1,\tau+\Delta\tau)} - M\theta_{(m,k-1,\tau+\Delta\tau)} + (1+4M)\theta_{(m,k,\tau+\Delta\tau)} = \theta_{(m,k,\tau)} \qquad (1.50)$$

Each equation of (1.50) contains 5 unknowns, therefore the set of simultaneous equations can no longer be solved in the simple manner used for the tri-diagonal system of the one-dimensional case. If the bar were divided into N space increments and the difference equations (1.49) had linear boundary conditions, it would be necessary to invert an $(N+1)\times(N+1)$ matrix at each time step. If the boundary

conditions are not linear, much more work would be required to obtain a solution. The backward difference method does have the advantage of unconditional stability, thus it may be possible to use large time steps.

(iii) Central difference

This method is the analogue of the Crank-Nicolson method of the one-dimensional case; the space derivatives are evaluated at $\tau+\frac{1}{2}\Delta\tau$. Taking an average for the evaluation of the space derivatives as used in equations (1.49) and (1.50):

$$\frac{\theta_{(n,k,\tau+\Delta\tau)}-\theta_{(n,k,\tau)}}{\Delta\tau}=\frac{1}{2}\left\{\frac{\theta_{(n+1,k,\tau)}-2\theta_{(n,k,\tau)}+\theta_{(n-1,k,\tau)}}{\Delta X^2}+\frac{\theta_{(n,k+1,\tau)}-2\theta_{(n,k,\tau)}+\theta_{(n,k-1,\tau)}}{\Delta Y^2}\right.$$

$$\left.+\frac{\theta_{(n+1,k,\tau+\Delta\tau)}-2\theta_{(n,k,\tau+\Delta\tau)}+\theta_{(n-1,k,\tau+\Delta\tau)}}{\Delta X^2}+\frac{\theta_{(n,k+1,\tau+\Delta\tau)}-2\theta_{(n,k,\tau+\Delta\tau)}+\theta_{(n,k-1,\tau+\Delta\tau)}}{\Delta Y^2}\right\} \quad (1.51)$$

rearranging:

$$\left[\theta_{(n+1,k,\tau+\Delta\tau)}-2(2+s)\theta_{(n,k,\tau+\Delta\tau)}+\theta_{(n-1,k,\tau+\Delta\tau)}+\theta_{(n,k+1,\tau+\Delta\tau)}+\theta_{(n,k-1,\tau+\Delta\tau)}\right]$$

$$=-\left[\theta_{(n+1,k,\tau)}-2(2-s)\theta_{(n,k,\tau)}+\theta_{(n-1,k,\tau)}+\theta_{(n,k+1,\tau)}+\theta_{(n,k-1,\tau)}\right] \quad (1.52)$$

Equation (1.52) is implicit as is (1.50). More calculation is, however, involved in the solution of the simultaneous equations which are obtained. As with the Crank-Nicolson method, equation (1.52) should be more accurate than either the forward or backward difference methods as both the time and space derivatives are evaluated at the same point, $\tau+\frac{1}{2}\Delta\tau$.

(iv) Alternating direction implicit method

This method has no analogue in the one-dimensional case. It is implicit, unconditionally stable and the chief objective is to avoid the solution of simultaneous equations with five unknowns. The difference equations are made implicit in one direction and explicit in the other for odd time steps then vice-versa for even time steps.

Making the difference analogue of equation (1.42) implicit in the X direction and explicit in the Y direction:

$$\frac{\theta_{(n,k,\tau+\Delta\tau)}-\theta_{(n,k,\tau)}}{\Delta\tau} = \frac{\theta_{(n+1,k,\tau+\Delta\tau)}-2\theta_{(n,k,\tau+\Delta\tau)}+\theta_{(n-1,k,\tau+\Delta\tau)}}{\Delta X^2}$$

$$+ \frac{\theta_{(n,k+1,\tau)}-2\theta_{(n,k,\tau)}+\theta_{(n,k-1,\tau)}}{\Delta Y^2} \qquad (1.53)$$

and for time step $\tau+2\Delta\tau$:

$$\frac{\theta_{(n,k,\tau+2\Delta\tau)}-\theta_{(n,k,\tau+\Delta\tau)}}{\Delta\tau} = \frac{\theta_{(n+1,k,\tau+\Delta\tau)}-2\theta_{(n,k,\tau+\Delta\tau)}+\theta_{(n-1,k,\tau+\Delta\tau)}}{\Delta X^2}$$

$$+ \frac{\theta_{(n,k+1,\tau+2\Delta\tau)}-2\theta_{(n,k,\tau+2\Delta\tau)}+\theta_{(n,k-1,\tau+2\Delta\tau)}}{\Delta Y^2} \qquad (1.54)$$

rearranging equation (1.53):

$$\left[\theta_{(n-1,k,\tau+\Delta\tau)}-(2+s)\theta_{(n,k,\tau+\Delta\tau)}+\theta_{(n+1,k,\tau+\Delta\tau)}\right]$$

$$= -\left[\theta_{(n,k+1,\tau)}-(2-s)\theta_{(n,k,\tau)}+\theta_{(n,k-1,\tau)}\right] \qquad (1.55)$$

similarly with equation (1.54):

$$\left[\theta_{(n,k-1,\tau+2\Delta\tau)}-(2+s)\theta_{(n,k,\tau+2\Delta\tau)}+\theta_{(n,k+1,\tau+2\Delta\tau)}\right]$$

$$= -\left[\theta_{(n-1,k,\tau+\Delta\tau)}-(2-s)\theta_{(n,k,\tau+\Delta\tau)}+\theta_{(n+1,k,\tau+\Delta\tau)}\right] \qquad (1.56)$$

Thus there are three unknowns in each equation, consequently the set can be solved by the simple elimination techniques used for a tri-diagonal matrix.

Previous work on the solution of two dimensional heat conduction problems by various difference equations has been done by Douglas (9) and Peaceman and Rachford (31). The work of Douglas is mainly of a theoretical nature but that of Peaceman and Rachford discusses the practical aspects.

H. Three dimensional problems

As extra space dimensions are considered, the problems become more complex. If an explicit method were to be used, the time steps would have to be very small because of an even more stringent stability requirement than that for two dimensional problems. When an implicit method is used say for the transient temperature distribution in a cube when subject to convection boundary conditions and N space increments are used, this would require the solution of $(N + 1)^3$ simultaneous equations at each time step, each equation having 7 unknowns. If N = 5, it would be necessary to solve 216 simultaneous equations at each time step!

The alternating direction implicit method is applicable in three space dimensions but it can become unstable. The method of Allada and Quon (1), (see section L) may be a little more satisfactory.

J. Truncation error

Equations (1.18) and (1.20) are only approximations to the partial derivatives since terms involving higher orders of ΔX and $\Delta \tau$ have been ignored. In order to obtain difference approximations which approximate the differential equations closely, examination of equations (1.14) and (1.15) show that if ΔX is small, generally, a better approximation will be obtained. It can also be seen that if the values of the higher order partial derivatives are small, a larger value of ΔX is permissible.

If a problem is considered in which no discontinuities exist e.g. the solution of the one-dimensional heat conduction equation for a plate with the boundary temperatures fixed and the initial temperature distribution a parabola in X, using the explicit method and considering the difference representations (1.14) and (1.15) leads to:

$$\left(\frac{\partial^2\theta}{\partial X^2}\right)_{(n,\tau)} = \frac{\theta_{(n+1,\tau)} - 2\theta_{(n,\tau)} + \theta_{(n-1,\tau)}}{\Delta X^2} - \frac{\Delta X^2}{12}\left(\frac{\partial^4\theta}{\partial X^4}\right)_{(n,\tau)} \qquad (1.57)$$

when terms involving orders higher than ΔX^2 are ignored. A similar analysis in the τ direction gives:

$$\left(\frac{\partial\theta}{\partial\tau}\right)_{(n,\tau)} = \frac{\theta_{(n,\tau+\Delta\tau)} - \theta_{(n,\tau)}}{\Delta\tau} - \frac{\Delta\tau}{2}\left(\frac{\partial^2\theta}{\partial\tau^2}\right)_{(n,\tau)} \qquad (1.58)$$

Equating equations (1.52) and (1.53) then rearranging:

$$\theta_{(m,\tau+\Delta\tau)} = \theta_{(m,\tau)} + M\left(\theta_{(m+1,\tau)} - 2\theta_{(m,\tau)} + \theta_{(m-1,\tau)}\right)$$
$$- \Delta\tau \cdot \frac{\Delta X^2}{12}\left(\frac{\partial^4\theta}{\partial X^4}\right)_{(m,\tau)} + \frac{\Delta\tau^2}{2}\left(\frac{\partial^2\theta}{\partial\tau^2}\right)_{(m,\tau)} \qquad (1.59)$$

Using the identity $$\frac{\partial^2\theta}{\partial\tau^2} = \frac{\partial^4\theta}{\partial X^4} \qquad (1.60)$$

it is obvious that equation (1.59) will be most accurate (i.e. the terms involving fourth order derivatives cancel out) when M = 1/6. This is a special case of the explicit method.

When there is a derivative boundary condition, the truncation error is dependent on the approximation at the boundary. Subtracting equation (1.31) from (1.30) gives:

$$\frac{\theta_{(1,\tau)} - \theta_{(-1,\tau)}}{2\Delta X} = \left(\frac{\partial\theta}{\partial X}\right)_{(0,\tau)} - \frac{\Delta X^3}{6}\left(\frac{\partial^3\theta}{\partial X^3}\right)_{(0,\tau)} \qquad (1.61)$$

It is seen, therefore, that the truncation error is dependent on the terms which contain the third and higher derivatives whereas in the case of a fixed boundary temperature it was the terms containing the fourth and higher derivatives.

If the values of the higher order derivatives were known, it would be quite simple to determine the number of space increments it was necessary to use and the required time step for a given error. Kardas (24) has worked on this basis in considering a plate with convective and adiabatic boundary conditions. He assumes the fourth order terms are those which control the accuracy of the difference approximation and estimates these from the known analytical solution,

presenting graphs for the truncation error as a function of Bi and τ for both the convective and insulated boundaries. It is noticed, however, that he does not take into account the different approximation at the boundary to that in the interior of the plate.

Kardas uses equations (1.57) and (1.58) to evaluate the truncation error which is given by:-

$$T.E. = -\left\{ \frac{\Delta X^2}{12}\left(\frac{\partial^4\theta}{\partial X^4}\right)_{(n,\tau)} - \frac{M\Delta X^2}{2}\left(\frac{\partial^4\theta}{\partial X^4}\right)_{(n,\tau)} \right\} \tag{1.62}$$

$$= -\frac{1}{4}(1-6M)\left\{ \frac{\Delta X^2}{3}\left(\frac{\partial^4\theta}{\partial X^4}\right)_{(n,\tau)} \right\} \tag{1.63}$$

at the interior points in the plate but at the boundary, equation (1.63) does not hold. From equation (1.14) it is seen that at the boundary X = 0:

$$\left(\frac{\partial^2\theta}{\partial X^2}\right)_{(0,\tau)} = \frac{2\left\{\theta_{(1,\tau)} - \theta_{(0,\tau)} - \Delta X\left(\frac{\partial\theta}{\partial X}\right)_{(0,\tau)} - \frac{\Delta X^3}{6}\left(\frac{\partial^3\theta}{\partial X^3}\right)_{(0,\tau)} - \frac{\Delta X^4}{24}\left(\frac{\partial^4\theta}{\partial X^4}\right)_{(0,\tau)}\right\}}{\Delta X^2} \tag{1.64}$$

rearranging and substituting $(\partial\theta/\partial X)_{(0,\tau)} = Bi\,\theta_{(0,\tau)}$ leads to:

$$\left(\frac{\partial^2\theta}{\partial X^2}\right)_{(0,\tau)} = \frac{2\theta_{(1,\tau)} - 2\theta_{(0,\tau)} - 2\Delta X Bi\,\theta_{(0,\tau)}}{\Delta X^2} - \frac{\Delta X}{3}\left(\frac{\partial^3\theta}{\partial X^3}\right)_{(0,\tau)} - \frac{\Delta X^2}{12}\left(\frac{\partial^4\theta}{\partial X^4}\right)_{(0,\tau)} \tag{1.65}$$

Therefore the truncation error is:

$$\frac{M\Delta X^2}{2}\left(\frac{\partial^4\theta}{\partial X^4}\right)_{(0,\tau)} - \frac{\Delta X^2}{12}\left(\frac{\partial^4\theta}{\partial X^4}\right)_{(0,\tau)} - \frac{\Delta X}{3}\left(\frac{\partial^3\theta}{\partial X^3}\right)_{(0,\tau)} \tag{1.66}$$

Obviously, more error is introduced at the boundary than at the interior points in the plate when the heat conduction equation is represented by finite difference approximations in the usual manner.

In practical problems, however, it is not usually

possible to know the values of the higher order derivatives and the determination of the lattice density is largely a matter of intuition.

K. Stability

The problem of stability has been mentioned in earlier sections. In this section the problem will be considered in greater detail.

The use of a difference equation, as shown above, will introduce errors. Also, the actual rounding operations on the computer will produce errors. If the error introduced at one time level by a difference formulation becomes greater at the next time level by the use of the difference equation, the difference equation is unstable because errors will accumulate. If errors introduced remain constant or become smaller, generally the difference equation is stable.

Considering equation (1.18) in which there are corresponding errors $E_{(n,\tau)}$ in $\theta_{(n,\tau)}$:

$$(\theta_{(n,\tau+\Delta\tau)} + E_{(n,\tau+\Delta\tau)}) = (\theta_{(n,\tau)} + E_{(n,\tau)}) + M[(\theta_{(n+1,\tau)} + E_{(n+1,\tau)}) - 2(\theta_{(n,\tau)} + E_{(n,\tau)}) + (\theta_{(n-1,\tau)} + E_{(n-1,\tau)})] \quad (1.67)$$

Thus it is seen by examination of equation (1.67) that an equation equivalent to (1.18) can be written for the error propagation:

$$E_{(n,\tau+\Delta\tau)} = E_{(n,\tau)} + M(E_{(n-1,\tau)} - 2E_{(n,\tau)} + E_{(n+1,\tau)}) \quad (1.68)$$

Therefore, the errors at any mesh point are propagated forward according to equation (1.68). The difference equation for the error propagation takes the same form as the difference equation for the function whenever the original differential

equation and its boundary conditions are linear.

O'Brien et al (30) present the theory of stability analysis. Briefly, they explain, by the use of Fourier series, that the error can be represented by a function:

$$E(X,\tau) = e^{\delta\tau} e^{i\kappa x} \tag{1.69}$$

Substituting this function into equation (1.18):

$$e^{\delta\tau} e^{\delta\Delta\tau} e^{i\kappa x} = e^{\delta\tau} e^{i\kappa x} + M(e^{\delta\tau} e^{i\kappa x} e^{i\kappa\Delta x} - 2e^{\delta\tau} e^{i\kappa x} + e^{\delta\tau} e^{i\kappa x} e^{-i\kappa\Delta x}) \tag{1.70}$$

$$\therefore \xi - 1 = M(e^{i\kappa\Delta x} - 2 + e^{-i\kappa\Delta x}) \tag{1.71}$$

$$= 2M(\cos(\kappa\Delta X) - 1) \tag{1.72}$$

$$\xi = 1 - 4M\sin^2(\kappa\Delta X/2) \tag{1.73}$$

for stability $|\xi| \leq 1$

$$\therefore -1 \leq [1 - 4M\sin^2(\kappa\Delta X/2)] \leq 1 \tag{1.74}$$

It can be seen from equation (1.74) that for stability $M \leq \frac{1}{2}$

Fig. 1.4, first presented by Richtmeyer (34), when the explicit method is used with M = 0.55 for the problem:

$$\tau = 0 \quad 0 \leq X \leq 1 \quad \theta_{(n,0)} = X \tag{1.75}$$

$$\tau > 0 \quad X = 1 \quad \left(\frac{\partial\theta}{\partial X}\right)_{(N,\tau)} = 0$$

illustrates instability.

It is readily seen that the approximate solution diverges rapidly from the analytical solution as τ increases.

Application of the above stability analysis to the Crank-Nicolson and backward difference methods shows them to be unconditionally stable, i.e. there is no restriction on M.

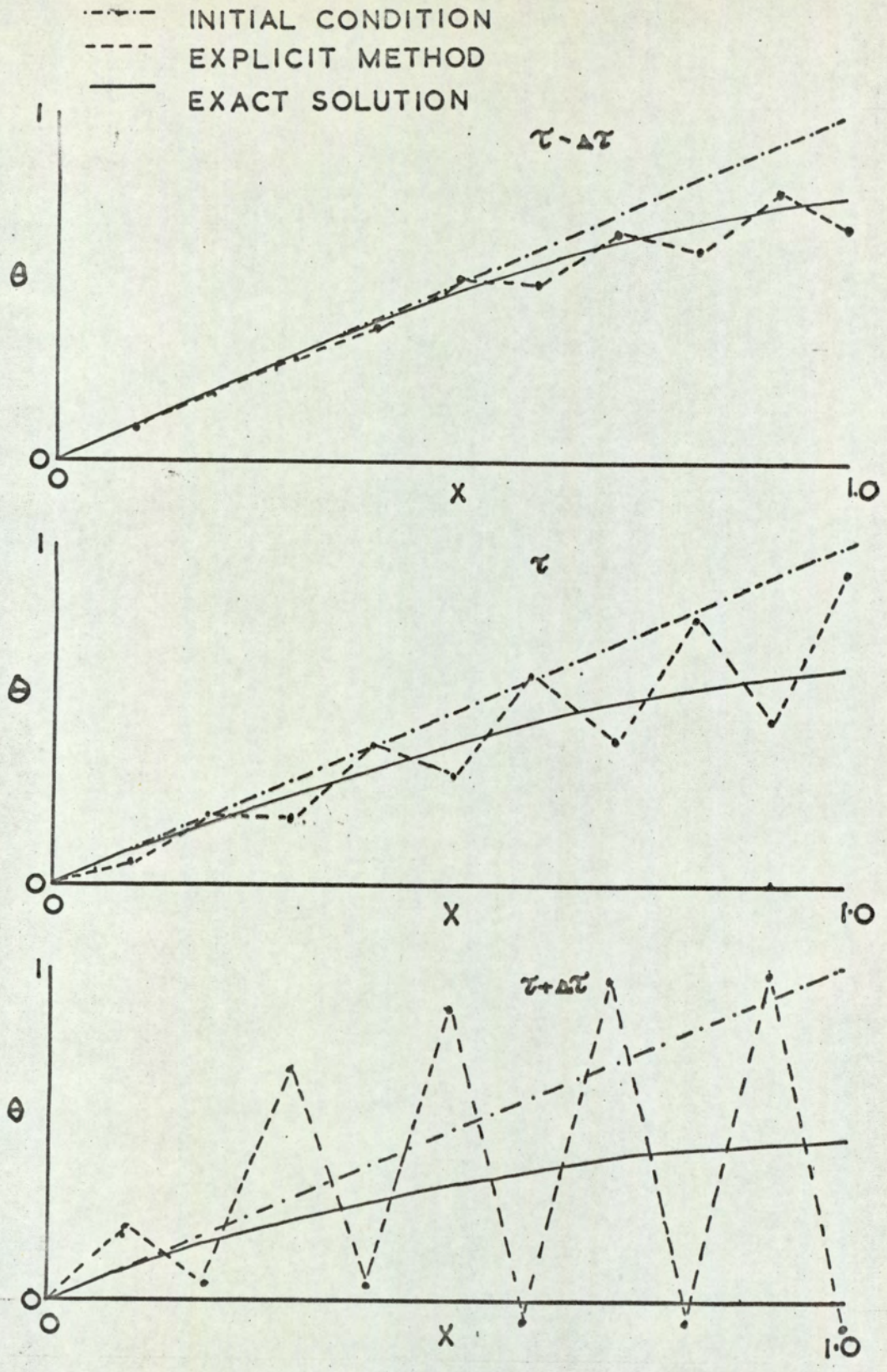


FIG 1.4 INSTABILITY OF THE EXPLICIT METHOD

The method of stability analysis presented by O'Brien et al is useful in the case where the boundary conditions are fixed but in the case of derivative boundary conditions this method breaks down. A second method, the matrix method, does hold.

Consider the use of the explicit method for a plate with convective and adiabatic boundary conditions:

Equation (1.18) can be rearranged:

$$\theta_{(n,\tau+\Delta\tau)} = M\theta_{(n+1,\tau)} + (1-2M)\theta_{(n,\tau)} + M\theta_{(n-1,\tau)} \tag{1.76}$$

and the two boundary equations:

$$\theta_{(0,\tau+\Delta\tau)} = M\theta_{(1,\tau)} + (1-2M)\theta_{(0,\tau)} + M(\theta_{(1,\tau)} - 2\Delta X Bi\,\theta_{(0,\tau)}) \tag{1.77}$$

when X = 0

and:

$$\theta_{(N,\tau+\Delta\tau)} = M\theta_{(N-1,\tau)} + (1-2M)\theta_{(N,\tau)} + M\theta_{(N-1,\tau)} \tag{1.78}$$

when X = 1.

If say 5 increments are used, the equations can be expressed in matrix form:

$$\begin{bmatrix} \theta_{(0,\tau+\Delta\tau)} \\ \theta_{(1,\tau+\Delta\tau)} \\ \theta_{(2,\tau+\Delta\tau)} \\ \theta_{(3,\tau+\Delta\tau)} \\ \theta_{(4,\tau+\Delta\tau)} \\ \theta_{(5,\tau+\Delta\tau)} \end{bmatrix} = \begin{bmatrix} [(1-2M)-2\Delta X Bi M] & 2M & & & & \\ M & (1-2M) & M & & & \\ & M & (1-2M) & M & & \\ & & M & (1-2M) & M & \\ & & & M & (1-2M) & M \\ & & & & 2M & (1-2M) \end{bmatrix} \begin{bmatrix} \theta_{(0,\tau)} \\ \theta_{(1,\tau)} \\ \theta_{(2,\tau)} \\ \theta_{(3,\tau)} \\ \theta_{(4,\tau)} \\ \theta_{(5,\tau)} \end{bmatrix} \tag{1.79}$$

$$\text{i.e } \theta_{(n,\tau+\Delta\tau)} = A\,\theta_{(n,\tau)} \qquad (1.80)$$

For stability, the maximum eigenvalue of A must be less than or equal to unity.

It can be seen from equation (1.79) that the value of Bi will have an effect on stability. Whereas O'Brien et al conclude that the Crank-Nicolson method is unconditionally stable, Keast and Mitchell (25) observe that in some circumstances under derivative boundary conditions, the Crank-Nicolson method does exibit instability. It appears that the instability which arises in the Crank-Nicolson method is in the main limited to a persistent error. Keast and Mitchell also show that the explicit method which is stable for $M \leq 0.5$ when fixed temperature boundary conditions are used becomes unstable at lower values of M when there is a derivative boundary condition.

Dusinberre (12) realised that the explicit method was likely to be unstable for lower values of M and proposed that in the explicit method, negative coefficients of $\theta_{(n,\tau)}$, when calculating $\theta_{(n,\tau+\Delta\tau)}$, should be avoided. For this condition to be met, from equation (1.77):

$$M \leq \frac{1}{2 + 2\Delta x Bi} \qquad (1.81)$$

The concept of instability can be explained in a simpler manner. Considering equation (1.18), assuming there is the worst possible error distribution on the mesh points at the

current time level, i.e. at the three nodal points n - 1, n, n + 1, there are errors of $-E'$, $+E'$, $-E'$ respectively. Application of equation (1.63) gives:

$$E_{(n,\tau+\Delta\tau)} = E' + M(-E' - 2E' - E')$$

$$\therefore E_{(n,\tau+\Delta\tau)} = E'(1-4M) \qquad (1.82)$$

$$\therefore E_{(n,\tau+\Delta\tau)} \text{ is only } \leq |E'| \text{ provided } M \leq \tfrac{1}{2}$$

As has been shown, the explicit method is subject to very severe restrictions, that is $\Delta\tau$ has to be small to avoid error growth. Richardson (33) presented a method for the solution of the heat conduction equation which at first seemed attractive:

$$\frac{\theta_{(n,\tau+\Delta\tau)} - \theta_{(n,\tau-\Delta\tau)}}{2\Delta\tau} = \frac{\theta_{(n+1,\tau)} - 2\theta_{(n,\tau)} + \theta_{(n-1,\tau)}}{\Delta x^2} \qquad (1.83)$$

that is, the difference analogue of the time derivative is calculated using the previous as well as the future time step. This method has been shown to be unconditionally unstable. It is suggested by O'Brien et al that Richardson did not realise the instability of his method as it was used only up to a small value of τ. Consequently, the errors had insufficient time to accumulate.

L. Stable explicit methods

(i) Dufort and Frankel

The explicit method has the disadvantage of requiring a small time step and the implicit methods requiring the solution of simultaneous equations. These drawbacks may be overcome by the use of the Dufort and Frankel method. Their formulation is:

$$\frac{\theta_{(n,\tau+\Delta\tau)} - \theta_{(n,\tau-\Delta\tau)}}{2\Delta\tau} = \frac{\theta_{(n+1,\tau)} - \theta_{(n,\tau+\Delta\tau)} - \theta_{(n,\tau-\Delta\tau)} + \theta_{(n-1,\tau)}}{\Delta x^2} \qquad (1.84)$$

Equation (1.84) is explicit but has the disadvantage of being unable to use a progressively increasing time step since values of θ are required at the previous as well as at the current time levels. It is also necessary to use another method to calculate the temperatures at the first time step before the Dufort and Frankel method can be used. Use of the backward difference method for the first time step would seem reasonable.

(ii) Alternating direction explicit

The Dufort and Frankel method has the disadvantage of requiring the temperatures at two time levels to calculate those at a third. Allada and Quon (1) present the alternating direction explicit method which is claimed to be stable for all values of M. Their presentation is for two and three

dimensional problems but there is an analogue in the one-dimensional case. The basis of the method is to use the temperatures at the future time level as soon as they are calculated in the direction of increasing n at one time level and to work in the direction of decreasing n for the next time step. The resulting one-dimensional difference equation, assuming the temperatures are evaluated in order of increasing n is:

$$\frac{\theta_{(n,\tau+\Delta\tau)} - \theta_{(n,\tau)}}{\Delta\tau} = \frac{\theta_{(n-1,\tau+\Delta\tau)} - \theta_{(n,\tau+\Delta\tau)} - \theta_{(n,\tau)} + \theta_{(n+1,\tau)}}{\Delta x^2} \qquad (1.85)$$

rearranging:

$$\theta_{(n,\tau+\Delta\tau)} = \frac{\theta_{(n,\tau)}}{(1+M)} + \frac{M}{(1+M)}\left(\theta_{(n-1,\tau+\Delta\tau)} - \theta_{(n,\tau)} + \theta_{(n+1,\tau)}\right) \qquad (1.86)$$

The corresponding equation for the error propagation is:

$$E_{(n,\tau+\Delta\tau)} = \frac{E_{(n,\tau)}}{(1+M)} + \frac{M}{(1+M)}\left(E_{(n-1,\tau+\Delta\tau)} - E_{(n,\tau)} + E_{(n+1,\tau)}\right) \qquad (1.87)$$

The stability of equation (1.86) was tested by following the propagation of an initial error distribution as shown in table 1.5. It is assumed that the error distribution is 0, 1, -1, 1, -1, 0 at time τ. Table 1.5 shows the decay of the errors with time. It is interesting to note that the application of an alternating direction procedure tends to spread the errors over all the nodal points. If the procedure had been used in the same direction for each time step, the errors would tend to be larger when n = 1.

TABLE 1.5

ERROR PROPAGATION USING A.D.E.P.

N = 5 M = 2

τ	NODE ERROR					
	0	1	2	3	4	5
τ	0	1	-1	1	-1	0
$\tau + \Delta\tau$	0	- 1	0.3333	- 0.7777	- 0.1852	0
$\tau + 2\Delta\tau$	0	- 0.2638	-0.8958	- 0.1771	0.4567	0
$\tau + 3\Delta\tau$	0	- 0.5092	-0.1589	- 0.3847	- 0.1042	0

Error distribution at $\tau + 2\Delta\tau$ proceeding in the same direction (n increasing) as at $\tau + \Delta\tau$ is:

$\tau + 2\Delta\tau$	0	0.5556	-0.2593	- 0.0370	- 0.0371	0

Equation (1.86) relies on the availability of a fixed boundary temperature. Using the alternating direction explicit method for a plate with convection and adiabatic boundary conditions at the boundary X = 0 leads to:

$$\theta_{(0,\tau+\Delta\tau)} = \frac{\theta_{(0,\tau)}}{(1+M)} + \frac{M}{(1+M)}\left(\theta_{(-1,\tau+\Delta\tau)} - \theta_{(0,\tau)} + \theta_{(1,\tau)}\right) \qquad (1.88)$$

an approximation is required for $\theta_{(-1,\tau+\Delta\tau)}$; this can be obtained from equation (1.35):

$$\theta_{(-1,\tau+\Delta\tau)} = \theta_{(1,\tau+\Delta\tau)} - 2\Delta X\, Bi\, \theta_{(0,\tau+\Delta\tau)} \qquad (1.89)$$

Substitution leads to:

$$\theta_{(0,\tau+\Delta\tau)} = \frac{\theta_{(0,\tau)}}{(1+M)} + \frac{M}{(1+M)}\left(\theta_{(1,\tau+\Delta\tau)} - 2\Delta X\, Bi\, \theta_{(0,\tau+\Delta\tau)} - \theta_{(0,\tau)} + \theta_{(1,\tau)}\right) \qquad (1.90)$$

which is implicit.

To overcome this restriction, the formulation used in the explicit method could be used at n = 0. This leads to:

$$\theta_{(0,\tau+\Delta\tau)} = \theta_{(0,\tau)} + M\left(2\theta_{(1,\tau)} - 2\theta_{(0,\tau)} - 2\Delta X\, Bi\, \theta_{(0,\tau)}\right) \qquad (1.91)$$

If equation (1.91) were used when n = 0, proceeding in the direction of increasing n, it would be permissible to use equation (1.36) at the adiabatic face at $\tau+\Delta\tau$. This leads to:

$$\theta_{(N,\tau+\Delta\tau)} = \frac{\theta_{(N,\tau)}}{(1+M)} + \frac{M}{(1+M)}\left(\theta_{(N-1,\tau+\Delta\tau)} - \theta_{(N,\tau)} + \theta_{(N-1,\tau)}\right) \qquad (1.92)$$

Proceeding in the direction of decreasing n would require the usual explicit formulation to be used at the adiabatic face but this time the alternating direction explicit formulation can be made when X = 0.

Allada and Quon make numerical experiments with the alternating direction implicit and explicit methods for the three dimensional problem of a cube with step change and adiabatic boundary conditions. They make no mention of the approximation used for the boundary temperature at the adiabatic face. They conclude that neither of the above mentioned methods are particularly accurate but the explicit method is very much faster.

M. Solution of the simultaneous equations.

Once the partial derivatives in the heat conduction equation have been replaced by their difference analogues it is necessary to solve the resulting equations. The use of an implicit differencing technique results in a set of simultaneous equations which have to be solved by an elimination process.

In one-dimensional heat conduction problems the coefficients of the equations form a tri-diagonal matrix; taking the backward difference method as an example for a plate with convection and adiabatic boundary conditions, the set of equations can be written in matrix form:

$$\begin{bmatrix} [(1+2M)+2M\Delta X Bi] & -2M & & & \\ -M & (1+2M) & -M & & \\ & -M & (1+2M) & -M & \\ & & -M & (1+2M) & -M \\ & & & -M & (1+2M)-M \\ & & & & -2M \quad (1+2M) \end{bmatrix} \begin{bmatrix} \theta_{(0,\tau+\Delta\tau)} \\ \theta_{(1,\tau+\Delta\tau)} \\ \theta_{(2,\tau+\Delta\tau)} \\ \theta_{(3,\tau+\Delta\tau)} \\ \theta_{(4,\tau+\Delta\tau)} \\ \theta_{(5,\tau+\Delta\tau)} \end{bmatrix} = \begin{bmatrix} \theta_{(0,\tau)} \\ \theta_{(1,\tau)} \\ \theta_{(2,\tau)} \\ \theta_{(3,\tau)} \\ \theta_{(4,\tau)} \\ \theta_{(5,\tau)} \end{bmatrix} \quad (1.93)$$

when 5 space increments are used.

Equations (1.93) have to be solved at each time step.

The solution of equations (1.93) is simple. The method as outlined in Modern Computing Methods (29) is as follows.

Equations (1.93) are analogous to the general set of equations:

$$\begin{bmatrix} b_0 & c_0 & & & & & \\ a_1 & b_1 & c_1 & & & & \\ & a_2 & b_2 & c_2 & & & \\ & & a_3 & b_3 & c_3 & & \\ & & & a_4 & b_4 & c_4 & \\ & & & & a_5 & b_5 & \end{bmatrix} \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix} = \begin{bmatrix} d_0 \\ d_1 \\ d_2 \\ d_3 \\ d_4 \\ d_5 \end{bmatrix} \qquad (1.94)$$

Using the first equation of (1.94) to eliminate y_0 from the second equation by multiplying the first equation by a_1/b_0 and subtracting from the second.

$$b_0 y_0 + c_0 y_1 = d_0 \qquad (1.95)$$

$$a_1 y_0 + b_1 y_1 + c_1 y_2 = d_1 \qquad (1.96)$$

which leads to:

$$\left(b_1 - \frac{a_1 c_0}{b_0}\right) y_1 + c_1 y_2 = \left(d_1 - \frac{a_1 d_0}{b_0}\right) \qquad (1.97)$$

i.e. $$\beta_1 y_1 + c_1 y_2 = \delta_2 \qquad (1.98)$$

The final equation of the set is:

$$\beta_5 y_5 = \delta_5 \qquad (1.99)$$

hence y_5 is calculated and back substitution into the general equation:

$$y_{n-1} = (\delta_{n-1} - c_n y_n) / \beta_{n-1} \qquad (1.100)$$

allows the remaining values of y_n to be calculated. Values of δ_n and β_n are calculated from:

$$\beta_n = b_n - \frac{a_n c_{n-1}}{\beta_{n-1}} \tag{1.101}$$

$$\delta_n = d_n - \frac{a_n \delta_{n-1}}{\beta_{n-1}} \tag{1.102}$$

and $\beta_0 = b_0$, $\delta_0 = d_0$

When there is a radiation boundary condition, the first equation of (1.94) is non-linear hence an iterative technique has to be applied. Again, taking the backward difference method as an example the equations can be written:

$$\begin{bmatrix} [(1+2M)+2M\Delta X R\theta^3_{(0,\tau+\Delta\tau)}] & -2M & & & & & \\ -M & (1+2M) & -M & & & & \\ & -M & (1+2M) & -M & & & \\ & & -M & (1+2M) & -M & & \\ & & & -M & (1+2M) & -M & \\ & & & & -2M & (1+2M) & \end{bmatrix} \begin{bmatrix} \theta_{(0,\tau+\Delta\tau)} \\ \theta_{(1,\tau+\Delta\tau)} \\ \theta_{(2,\tau+\Delta\tau)} \\ \theta_{(3,\tau+\Delta\tau)} \\ \theta_{(4,\tau+\Delta\tau)} \\ \theta_{(5,\tau+\Delta\tau)} \end{bmatrix} = \begin{bmatrix} \theta_{(0,\tau)} \\ \theta_{(1,\tau)} \\ \theta_{(2,\tau)} \\ \theta_{(3,\tau)} \\ \theta_{(4,\tau)} \\ \theta_{(5,\tau)} \end{bmatrix} \tag{1.103}$$

It is seen that $\theta_{(0,\tau+\Delta\tau)}$ is contained in the matrix of coefficients for $\theta_{(n,\tau+\Delta\tau)}$. Equations (1.103) can be solved by making an estimate for $\theta_{(0,\tau+\Delta\tau)}$ and performing the elimination mentioned above then comparing the estimate for $\theta_{(0,\tau+\Delta\tau)}$ with the new value obtained. Iteration is continued until the differences between two successive iterates for $\theta_{(0,\tau+\Delta\tau)}$ are within a desired convergence limit.

Another method which appears popular is the Newton-Raphson iteration (26). The set of difference equations obtained from the backward difference method for a plate with radiation and adiabatic boundary conditions may be expanded

in a Taylor series, truncated after the first partial derivatives, about an initial estimate of the temperatures $\theta_{(n,\tau+\Delta\tau)}$. This expansion leads to a set of linear simultaneous equations for $\Delta\theta^{(m)}_{(n,\tau+\Delta\tau)} = \theta^{(m+1)}_{(n,\tau+\Delta\tau)} - \theta^{(m)}_{(n,\tau+\Delta\tau)}$

$$\begin{bmatrix} [(1+2M)+8M\Delta x R\theta^{(m)3}_{(0,\tau+\Delta\tau)}] & -2M & & & & \\ -M & (1+2M) & -M & & & \\ & -M & (1+2M) & -M & & \\ & & -M & (1+2M) & -M & \\ & & & -M & (1+2M) & -M \\ & & & & -2M & (1+2M) \end{bmatrix} \begin{bmatrix} \Delta\theta^{(m)}_0 \\ \Delta\theta^{(m)}_1 \\ \Delta\theta^{(m)}_2 \\ \Delta\theta^{(m)}_3 \\ \Delta\theta^{(m)}_4 \\ \Delta\theta^{(m)}_5 \end{bmatrix} = \begin{bmatrix} -f_0 \\ -f_1 \\ -f_2 \\ -f_3 \\ -f_4 \\ -f_5 \end{bmatrix} \qquad (1.104)$$

where for instances, $f_1 = \theta_{(1,\tau)} + M\theta^{(m)}_{(0,\tau+\Delta\tau)} - (1+2M)\theta^{(m)}_{(1,\tau+\Delta\tau)} + M\theta^{(m)}_{(2,\tau+\Delta\tau)}$

and $\theta^{(0)}_{(n,\tau+\Delta\tau)} = \theta_{(n,\tau)}$

The above tri-diagonal system may be solved easily using the elimination method and iteration is continued until $\max | \Delta\theta^{(m)}_{(n,\tau+\Delta\tau)} |$ is within the desired convergence limit.

It is apparent that the Newton-Raphson iteration involves more calculation than the simple iteration method but it is possible that this method may be faster to converge.

Matrix iterative methods

The common iterative methods for solution of sets of simultaneous equations are: Jacobian, Gauss-Seidel and relaxation (accelerated Gauss-Seidel). All the methods consist of making initial estimates of the unknowns, then

calculating new (better) estimates by solving the rearranged equations. The three methods can be, and usually are, applied to sets of linear as well as non-linear equations. It would, however, be unnecessary to use an iterative method to obtain the solution of a set of linear simultaneous equations, the coefficients of which form a tri-diagonal matrix.

The simultaneous equations obtained using the backward difference method for a plate with radiation and adiabatic boundary conditions are written:

$$(1+2M)\theta_{(0,\tau+\Delta\tau)} + 2M\Delta X R\theta^4_{(0,\tau+\Delta\tau)} - 2M\theta_{(1,\tau+\Delta\tau)} = \theta_{(0,\tau)} \quad (1.105)$$

$$-M\theta_{(0,\tau+\Delta\tau)} + (1+2M)\theta_{(1,\tau+\Delta\tau)} - M\theta_{(2,\tau+\Delta\tau)} = \theta_{(1,\tau)} \quad (1.106)$$

$$\vdots$$

$$-2M\theta_{(4,\tau+\Delta\tau)} + (1+2M)\theta_{(5,\tau+\Delta\tau)} = \theta_{(5,\tau)} \quad (1.107)$$

The iterative methods used for the solution of equations (1.105) - (1.107) are:

(i) Jacobian

Equations (1.105) are rearranged to form the iteration algorithm:

$$\theta^{(m+1)}_{(0,\tau+\Delta\tau)} = \frac{\theta_{(0,\tau)} + 2M\theta^{(m)}_{(1,\tau+\Delta\tau)}}{(1+2M) + 2M\Delta X R\theta^{(m)3}_{(0,\tau+\Delta\tau)}} \quad (1.108)$$

$$\theta^{(m+1)}_{(1,\tau+\Delta\tau)} = \frac{\theta_{(1,\tau)} + M\theta^{(m)}_{(2,\tau+\Delta\tau)} + M\theta^{(m)}_{(0,\tau+\Delta\tau)}}{(1+2M)}$$

$$\vdots$$

$$\theta^{(m+1)}_{(5,\tau+\Delta\tau)} = \frac{\theta_{(5,\tau)} + 2M\theta^{(m)}_{(4,\tau+\Delta\tau)}}{(1+2M)}$$

The first equation of (1.108) is the same as that used

when elimination combined with simple iteration on the first equation only is used. If a large set of equations were being solved by this method, it can be seen that a large amount of computer store would be required since both the m^{th} and the $(m + 1)^{th}$ iterates must be stored simultaneously.

(ii) Gauss-Seidel

When this method is used, the new iterates are used as soon as they are calculated which means that only one set of iterates has to be stored at any one time. It would intuitively seem better to use the Gauss-Seidel method.

The iteration algorithms are:

$$\theta^{(m+1)}_{(0,\tau+\Delta\tau)} = \frac{\theta_{(0,\tau)} + 2M\theta^{(m)}_{(1,\tau+\Delta\tau)}}{(1+2M) + 2M\Delta x R \theta^{(m)3}_{(0,\tau+\Delta\tau)}} \qquad (1.109)$$

$$\theta^{(m+1)}_{(1,\tau+\Delta\tau)} = \frac{\theta_{(1,\tau)} + M\theta^{(m)}_{(2,\tau+\Delta\tau)} + M\theta^{(m+1)}_{(0,\tau+\Delta\tau)}}{(1+2M)} \qquad (1.110)$$

$$\vdots$$

$$\theta^{(m+1)}_{(5,\tau+\Delta\tau)} = \frac{\theta_{(5,\tau)} + 2M\theta^{(m+1)}_{(4,\tau+\Delta\tau)}}{(1+2M)}$$

The first equation of (1.109) is the same as the first equation of (1.108) since the $(m + 1)^{th}$ iterate for $\theta_{(1,\tau+\Delta\tau)}$ is not available.

(iii) Relaxation

This method is similar to the Gauss-Seidel except $\theta^{(m+1)}_{(n,\tau+\Delta\tau)}$ is re-named $\bar{\theta}^{(m+1)}_{(n,\tau+\Delta\tau)}$ and $\theta^{(m+1)}_{(n,\tau+\Delta\tau)}$ is

calculated:

$$\theta^{(m+1)}_{(n,\tau+\Delta\tau)} = \theta^{(m)}_{(n,\tau+\Delta\tau)} + \omega\left(\bar{\theta}^{(m+1)}_{(n,\tau+\Delta\tau)} - \theta^{(m)}_{(n,\tau+\Delta\tau)}\right) \qquad (1.111)$$

Relaxation methods were usually thought of as being useful only to hand computations because the choice of ω is usually intuitive. These methods are, however, becoming more popular with automatic computation.

Depending on the value of $\omega < 1$ or $\omega > 1$ relaxation methods are usually referred to as under - or over - relaxation. When $\omega = 1$, the relaxation method reduces to the Gauss-Seidel.

Matrix iterative methods are usually more popular with a larger number of equations, for example when a two dimensional system is under consideration. In this case the number of equations is $(N + 1)^2$. Iterative methods are generally preferred for the solution of two-dimensional heat conduction equations even though they may be linear. Peaceman and Rachford (31) consider the use of iterative methods for the solution of the two-dimensional transient heat conduction equation with linear boundary conditions. The solution of the Laplace equation is also considered. Reference is made to the work of Frankel (18) on the iterative solution of the Laplace equation. Varga (40) discusses and explains matrix iterative methods.

N. Integral method

The main work in this field appears to be that of Goodman (20). Schneider (38) presents the method applied to a plate with radiation and adiabatic boundary conditions and uniform initial temperature distribution. Fig. 1.5 shows the basis of the integral method for a plate. The temperature distribution through the plate is represented as a polynomial in X, the coefficients of which are a function of time. The problem is considered in two parts:

(a) $\tau \leq \tau_{x_p}$

(b) $\tau \geq \tau_{x_p}$

The degree of the polynomial is dependent upon the number of boundary conditions which are available. Two cases are considered:

(1) Quadratic Profile

(a) $\tau \leq \tau_{x_p}$

The temperature throughout the plate is represented by a parabola:

$$\theta = a_0 + a_1 X + a_2 X^2 \tag{1.112}$$

where a_0, a_1, and a_2 are functions of time

Integrating equation (1.7) with respect to X:

$$\int_0^{x_p} \frac{\partial \theta}{\partial \tau} dX = \int_0^{x_p} \frac{\partial^2 \theta}{\partial X^2} dX \tag{1.113}$$

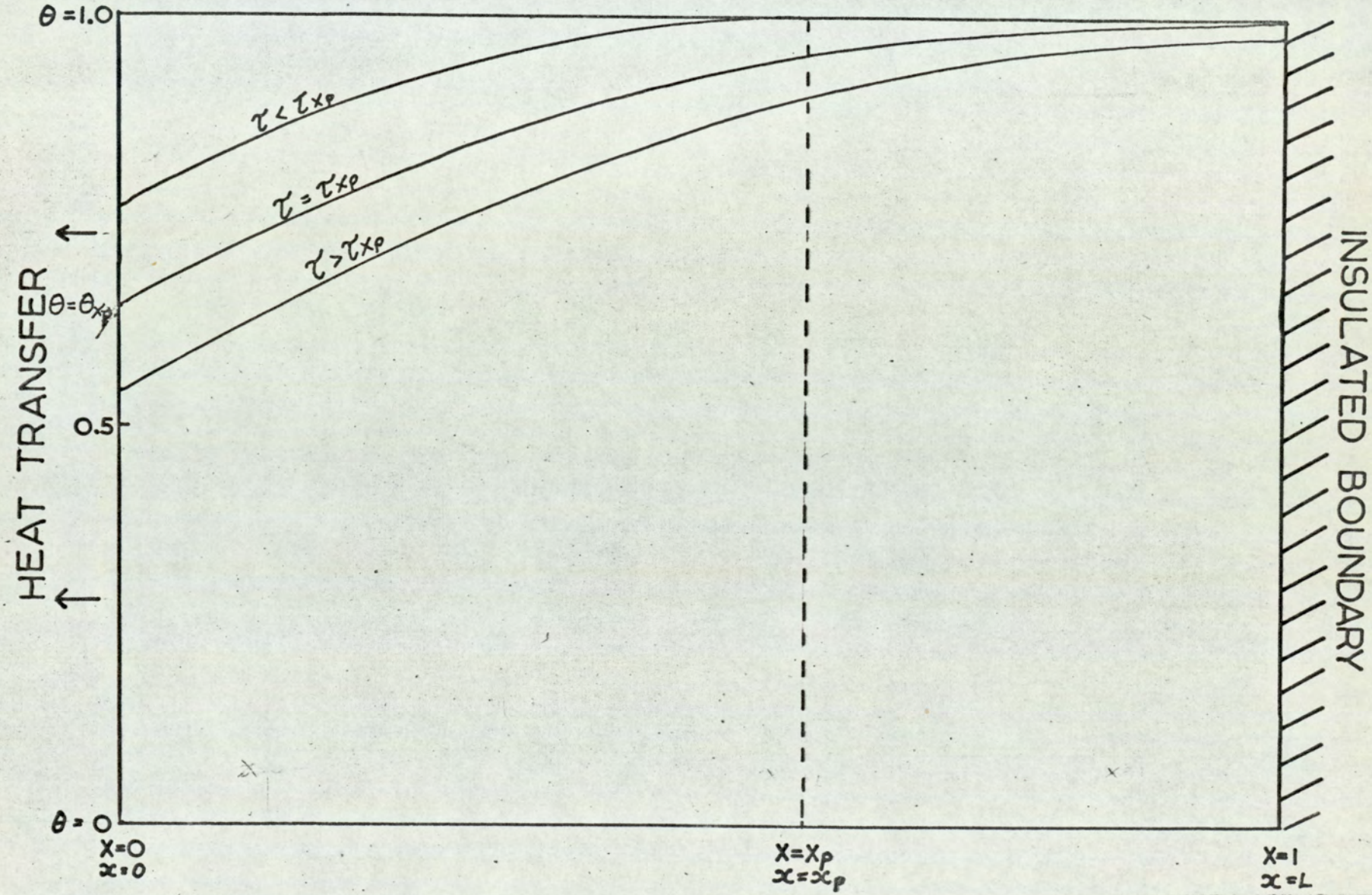


FIG 1.5 PRINCIPLE OF THE INTEGRAL METHOD FOR A PLATE

Using the rule for differentiating an integral:

$$\frac{d}{d\tau}\int_0^{X_p} \theta\, dX - \theta(X_p,\tau)\frac{dX_p}{d\tau} = -\frac{d\theta}{dX}(0,\tau) \tag{1.114}$$

$$\therefore \frac{d}{d\tau}\left[\int_0^{X_p} \theta\, dX - \theta(X_p,\tau)X_p\right] = -\frac{d\theta}{dX}(0,\tau) \tag{1.115}$$

Using the following boundary conditions:

$$X=0:\quad \frac{\partial\theta_0}{\partial X} = R\theta_0^4 = f\ (\text{say}) \tag{1.116}$$

$$X=X_p:\quad \theta(X_p,\tau) = 1 \tag{1.117}$$

$$\frac{\partial\theta}{\partial X}(X_p,\tau) = 0 \tag{1.118}$$

differentiating equation (1.112) with respect to X:

$$\frac{\partial\theta}{\partial X} = a_1 + 2a_2X$$

Substituting the boundary conditions leads to:

$$\theta = 1 - \frac{fX_p}{2} + fX - \frac{fX^2}{2X_p} \tag{1.119}$$

$\therefore$ when X = 0:

$$\theta_0 = 1 - \frac{fX_p}{2} \tag{1.120}$$

$$\therefore \quad X_p = \frac{2(1-\theta_0)}{f} = \frac{2(1-\theta_0)}{R\theta_0^4} \tag{1.121}$$

when $X_p = 1$ i.e. $x_p = L$:

$$\theta_{x_p}^3 = \frac{2}{R}\left[\frac{1}{\theta_{x_p}} - 1\right] \tag{1.122}$$

Substituting the profile for θ (given by equation (1.119)) into equation (1.115), performing the necessary integration and substituting $\theta(X_p,\tau) = 1$ gives:

$$\frac{1}{6}\frac{d}{d\tau}\left[fX_p^2\right] = f \tag{1.123}$$

From equation (1.121):

$$\int X_p^2 = R\theta_o^4 \left[\frac{2(1-\theta_o)}{R\theta_o^4}\right]^2$$

$$= \frac{4(1-\theta_o)^2}{R\theta_o^4}$$

$$\therefore \frac{2}{3}\frac{d}{d\tau}\left[\frac{(1-\theta_o)^2}{R\theta_o^4}\right] = R\theta_o^4 \qquad (1.124)$$

Performing the necessary differentiation to extract $d\theta_o/d\tau$:

$$-\frac{4}{3}\left[\theta_o(1-\theta_o) + 2(1-\theta_o)^2\right]\frac{d\theta_o}{d\tau} = R^2\theta_o^9 \qquad (1.125)$$

$$\therefore -\frac{4}{3}\left[\frac{2}{\theta_o^9} - \frac{3}{\theta_o^8} + \frac{1}{\theta_o^7}\right]d\theta_o = R^2 d\tau \qquad (1.126)$$

$$-\frac{4}{3}\int_1^{\theta_o}\left[\frac{2}{\theta_o^9} - \frac{3}{\theta_o^8} + \frac{1}{\theta_o^7}\right]d\theta_o = \int_0^{\tau} R^2 d\tau \qquad (1.127)$$

$$\therefore R^2\tau = -\frac{4}{3}\left[-\frac{2}{8\theta_o^8} + \frac{3}{7\theta_o^7} - \frac{1}{6\theta_o^6}\right]_1^{\theta_o} \qquad (1.128)$$

simplifying:

$$\tau = \frac{1}{R^2}\left\{\frac{4}{3\theta_o^8}\left[\frac{1}{6}\theta_o^2 - \frac{3}{7}\theta_o + \frac{1}{4}\right] + \frac{1}{63}\right\} \qquad (1.129)$$

Using equation (1.121) to substitute for f in equation (1.119) leads to:

$$\theta = \theta_o + 2(1-\theta_o)\left[\left(\frac{X}{X_p}\right) - \frac{1}{2}\left(\frac{X}{X_p}\right)^2\right] \qquad (1.130)$$

Therefore, for $\theta_{x_p} \leq \theta_o \leq 1$, τ can be calculated from equation (1.129), X_p from equation (1.121) and the temperature profile from equation (1.130). θ_{x_p} is obtained from equation (1.122)

(b) $\tau \geq \tau_{x_p}$

The temperature profile is represented by a new polynomial:

$$\theta = b_0 + b_1 X + b_2 X^2 \tag{1.131}$$

with the boundary conditions:

$$X = 0: \quad \frac{\partial \theta_0}{\partial X} = R\theta_0^4 = f \tag{1.132}$$

$$X = 1: \quad \frac{\partial \theta(1,\tau)}{\partial X} = 0 \tag{1.133}$$

Differentiating equation (1.131) leads to:

$$\frac{\partial \theta}{\partial X} = b_1 + 2 b_2 X \tag{1.134}$$

Substituting the boundary conditions:

$$\theta = b_0 + fX - \frac{fX^2}{2} \tag{1.135}$$

$$\int_0^1 \frac{\partial \theta}{\partial \tau} dX = \int_0^1 \frac{\partial^2 \theta}{\partial X^2} dX$$

since the limits of integration are fixed:

$$\frac{d}{d\tau}\int_0^1 \theta \, dX = -\frac{d\theta}{dX}(0,\tau) \tag{1.136}$$

performing the integration:

$$\frac{d}{d\tau}\left[b_0 + \frac{R\theta_0^4}{3}\right] = -R\theta_0^4 \tag{1.137}$$

from equation (1.135) when X = 0:

$$b_0 = \theta_0 \tag{1.138}$$

$$\therefore \frac{d}{d\tau}\left[\theta_0 + \frac{R\theta_0^4}{3}\right] = -R\theta_0^4 \tag{1.139}$$

performing the necessary differentiation:

$$\int_{\theta_{xp}}^{\theta_o} \left[\frac{1}{R\theta_o^4} + \frac{4}{3\theta_o}\right] d\theta_o = -\int_{\tau_{xp}}^{\tau} d\tau$$

$$\therefore \tau = \frac{1}{3}\left[\frac{1}{R}\left(\frac{1}{\theta_o^3} - \frac{1}{\theta_{xp}^3}\right) + 4\ln\left(\frac{\theta_{xp}}{\theta_o}\right)\right] + \tau_{xp} \qquad (1.140)$$

The temperature profile in the slab for $\tau \geq \tau_{xp}$ is given by:

$$\theta = \theta_o + R\theta_o^4 X - \frac{R\theta_o^4 X^2}{2} \qquad (1.141)$$

(2) Cubic profile

(a) $\tau \leq \tau_{xp}$

The temperature profile through the slab is represented by a cubic:

$$\theta = a_0 + a_1 X + a_2 X^2 + a_3 X^3 \qquad (1.142)$$

Thus it is necessary to have an extra boundary condition. This is:

$$\frac{\partial^2\theta}{\partial X^2}(1,\tau) = 0 \qquad (1.143)$$

i.e. since $\frac{\partial\theta}{\partial X}(1,\tau) = 0$ always,

the boundary condition as given by equation (1.143) is justified.

Differentiating equation (1.142):

$$\frac{\partial\theta}{\partial X} = a_1 + 2a_2 X + 3a_3 X^2 \qquad (1.144)$$

$$\frac{\partial^2\theta}{\partial X^2} = 2a_2 + 6a_3 X \qquad (1.145)$$

when X = 0 : $$\frac{\partial \theta_o}{\partial X} = R\theta_o^4 = f \tag{1.146}$$

$$X = X_p : \quad \frac{\partial \theta}{\partial X}(X_p, \tau) = 0 \tag{1.147}$$

$$\frac{\partial^2 \theta}{\partial X^2}(X_p, \tau) = 0 \tag{1.148}$$

$$\theta(X_p, \tau) = 1 \tag{1.149}$$

substituting the boundary conditions

$$\theta = 1 - \frac{fX_p}{3} + fX - \frac{fX^2}{X_p} + \frac{fX^3}{3X_p^2} \tag{1.150}$$

when X = 0, $\theta = \theta_o$

$$\therefore \quad \theta_o = 1 - \frac{fX_p}{3}$$

$$\therefore \quad X_p = \frac{3(1-\theta_o)}{f} = \frac{3(1-\theta_o)}{R\theta_o^4} \tag{1.151}$$

when $X_p = 1$:

$$\theta_{X_p}^3 = \frac{3}{R}\left[\frac{1}{\theta_{X_p}} - 1\right] \tag{1.152}$$

substituting the profile for into equation (1.115), performing the necessary integration and substitution of $\theta(X_p, \tau) = 1$ leads to:

$$\frac{1}{12}\frac{d}{d\tau}\left[fX_p^2\right] = f \tag{1.153}$$

$$\therefore \frac{3}{4}\frac{d}{d\tau}\left[\frac{(1-\theta_o)^2}{R\theta_o^4}\right] = R\theta_o^4 \tag{1.154}$$

extracting $d\theta_o/d\tau$:

$$-\frac{3}{2}\int_1^{\theta_o}\left[-\frac{3}{\theta_o^8} + \frac{1}{\theta_o^7} + \frac{2}{\theta_o^9}\right]d\theta_o = R^2\int_0^{\tau} d\tau \tag{1.155}$$

integrating and simplifying:

$$\tau = \frac{1}{R^2}\left[\frac{3}{2\theta_o^8}\left(\frac{1}{6}\theta_o^2 - \frac{3}{7}\theta_o + \frac{1}{4}\right) + \frac{1}{56}\right] \tag{1.156}$$

Using equation (1.151) to substitute for f in equation (1.150).

$$\theta = \theta_o + 3(1-\theta_o)\left[\left(\frac{X}{X_p}\right) - \left(\frac{X}{X_p}\right)^2 + \frac{1}{3}\left(\frac{X}{X_p}\right)^3\right] \qquad (1.157)$$

X_p can be calculated from equation (1.151) and τ and θ from equations (1.156) and (1.157) for $\theta_{xp} \leq \theta_o \leq 1$. θ_{xp} is obtained from equation (1.152).

(b) $\tau \geq \tau_{xp}$

$$\theta = b_o + b_1 X + b_2 X^2 + b_3 X^3 \qquad (1.158)$$

with the boundary conditions:

$$X=0: \quad \frac{\partial \theta_o}{\partial X} = R\theta_o^4 = f \qquad (1.159)$$

$$X=1: \quad \frac{\partial \theta}{\partial X}(1,\tau) = 0 \qquad (1.160)$$

$$\frac{\partial^2 \theta}{\partial X^2}(1,\tau) = 0 \qquad (1.161)$$

Differentiating equation (1.158)

$$\frac{\partial \theta}{\partial X} = b_1 + 2b_2 X + 3b_3 X^2$$

$$\frac{\partial^2 \theta}{\partial X^2} = 2b_2 + 6b_3 X$$

substituting the boundary conditions:

$$\theta = b_o + fX - fX^2 + \frac{f}{3}X^3 \qquad (1.162)$$

substituting the profile for θ into equation (1.136) and performing the integration.

$$\frac{d}{d\tau}\left[b_o + \frac{R\theta_o^4}{4}\right] = -R\theta_o^4 \qquad (1.163)$$

from equation (1.146) when X = 0, $b_o = \theta_o$

$$\therefore \frac{d}{d\tau}\left[\theta_o + \frac{R\theta_o^4}{4}\right] = -R\theta_o^4$$

performing the necessary differentiation, then integrating:

$$\int_{\theta_{xp}}^{\theta_o}\left[\frac{1}{R\theta_o^4} + \frac{1}{\theta_o}\right]d\theta_o = -\int_{\tau_{xp}}^{\tau} d\tau \qquad (1.164)$$

$$\therefore \quad \tau = \frac{1}{3R}\left[\frac{1}{\theta_o^3} - \frac{1}{\theta_{xp}^3}\right] + \ln\left(\frac{\theta_{xp}}{\theta_o}\right) + \tau_{xp} \qquad (1.165)$$

The temperature profile in the plate is then given by:

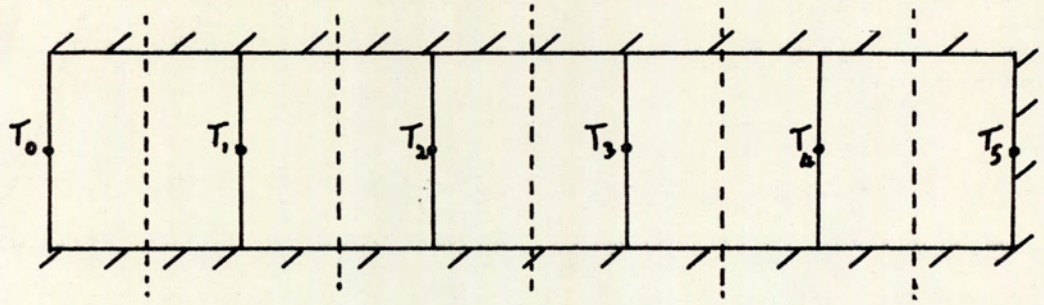
$$\theta = \theta_o + R\theta_o^4 X(1 - X + X^2/3) \qquad (1.166)$$

P. Analogue computer solution

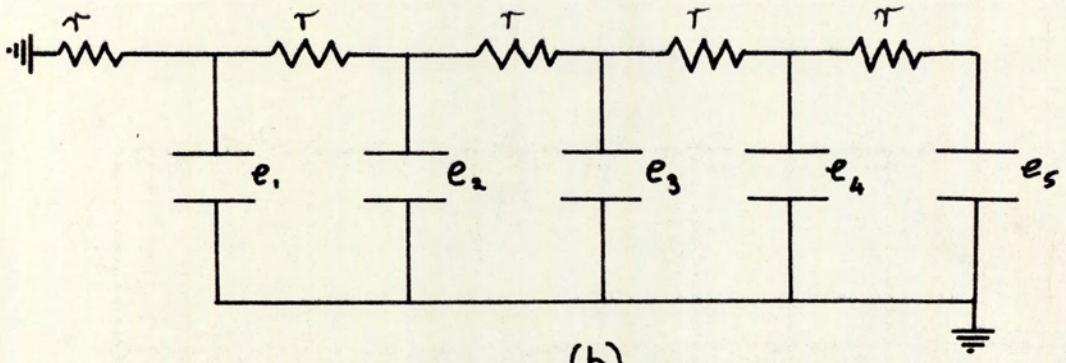
The analogue computer is useful for solving heat conduction problems. The general criterion for use of an analogue computer is whether a system can be simulated by an analogous electrical network. Consider a bar, which is insulated around its circumference and at one end, divided into equal sections along its length. The bar is at a uniform temperature throughout and the temperature at the open end is suddenly reduced to zero. This is a one-dimensional problem similar to that of a plate with step change and adiabatic boundary conditions. It is represented diagrammatically in fig 1.6 (a). It is assumed that each section of the bar is at a uniform temperature throughout. Each section behaves as a heat capacitor which interchanges heat with the next section due to a temperature difference between the two sections. An analogous electrical network is shown in fig. 1.6 (b). Initially the capacitors are charged to a certain potential and then at $t = 0$ C_1 is earthed through a resistance, r. The equation representing say the rate of change of potential on C_3 is:

$$C\frac{de_3}{dt} = \frac{e_4 - e_3}{r} - \frac{e_3 - e_2}{r} = \frac{e_4 - 2e_3 + e_2}{r} \qquad (1.167)$$

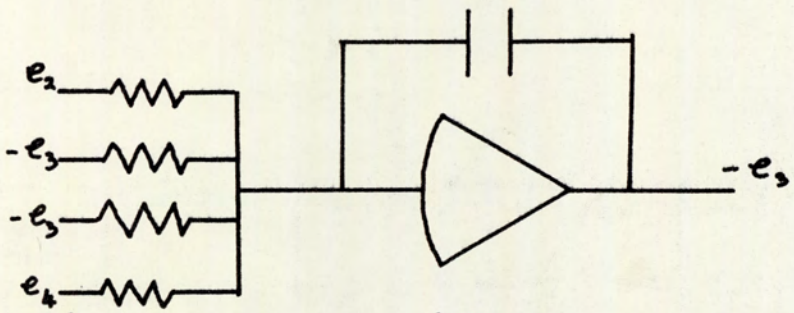
Equation (1.167) is analogous to the heat conduction equation which has been partially finite differenced:



(a)



(b)



(c)

FIG 1.6 ANALOGOUS ELECTRICAL NETWORK TO THE HEAT CONDUCTION EQUATION

$$\rho C_{p_3} \frac{dT_3}{dt} = \frac{k(T_4 - 2T_3 + T_2)}{\Delta x^2} \qquad (1.168)$$

where $C \propto C_p \Delta x / \rho$

$e_n \propto T_n$

$\tau \propto \Delta x / k$

Thus by choosing appropriate scaling factors the circuit shown in fig. 1.6 (b) can be made to behave in the same manner as the heat conductive circuit in fig. 1.6 (a).

Fig. 1.6 (b) is an impractical circuit as the capacitors would discharge due to leakage currents, hence it would be necessary to use an electrostatic voltmeter to measure the potential. To eliminate this difficulty, high gain amplifiers are used and a circuit which behaves in the same manner as the capacitor under consideration in fig. 1.6 (b) is shown in fig. 1.6 (c). The analogue computer symbol for 1.6 (c) is shown in fig. 1.7.

When there are derivative boundary conditions, these are easily represented using the circuits shown in figs. 1.8 (a) and (b). In fig. 1.8 (b) where there is a radiation boundary condition it is necessary to calculate θ_o^4 , that is a variable has to be squared twice. This is achieved using an "X^2 diode function generator", which approximates the square of a variable by a series of straight lines.

The use of the analogue computer for the radiation boundary problem has been considered by Zerkle and Sunderland who present graphically the temperature-time history of a

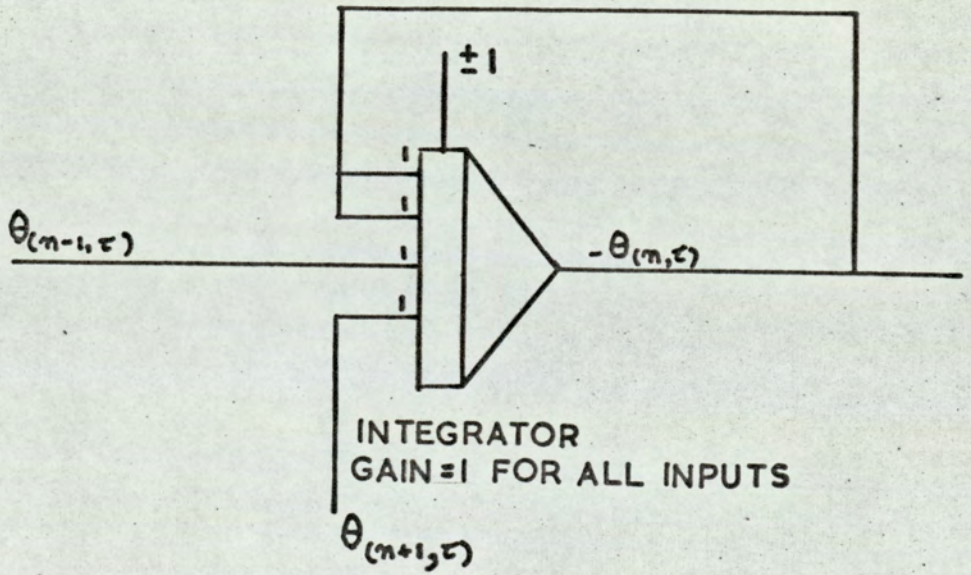
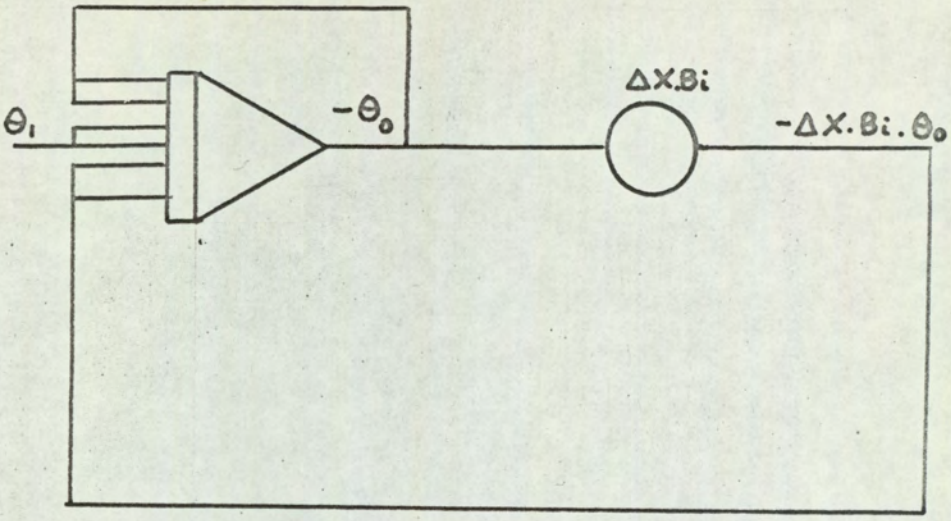
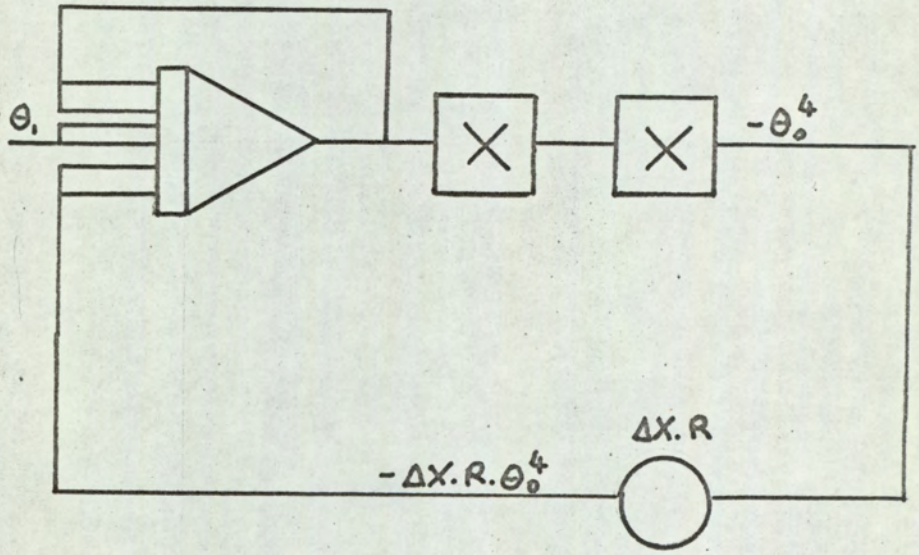


FIG 1.7 BASIC ANALOGUE COMPUTER MODULE FOR EQUATION 1.168 (time scaled by $1/\Delta X^2$)



(a) CONVECTION



(b) RADIATION

FIG 1.8 ANALOGUE COMPUTER CIRCUITS FOR DERIVATIVE BOUNDARY CONDITIONS.

plate for various values of R. The analogue computer has the advantage of providing a rapid solution. Since the value of R is set on a potentiometer it can be readily changed. Similarly it is simple to alter the initial condition. The main disadvantage of the analogue computer is the requirement of large amounts of equipment for all but the simplest problems; for example, if the solution of a two-dimensional problem is required it would be necessary to use 36 integrators whereas 6 may be sufficient in the one-dimensional case. If the physical properties of the material were temperature dependent a large amount of non-linear equipment would be required to simulate the problem.

The use of a hybrid computer for heat conduction problems is becoming more popular. There appears to be two ways: firstly use of an implicit finite difference method and use of the analogue section of the hybrid to solve the simultaneous equations which may be non-linear and, secondly, to partially finite difference the heat conduction equation and share a block of equipment for each node in turn, as all the nodes, except the one at the boundary, are similar. Work on these lines has been carried out by Electronic Associates Ltd. (14)

Q. Analytical Solutions

One technique for determining which method is best for the solution of a heat conduction problem is to compare the results obtained using approximate methods with the analytical solution for that problem. In practice, however, analytical solutions only exist for a small number of cases. Comparison of the approximate solutions with the analytical solution for the step change and convection boundary conditions gives an indication of the errors to be expected in the radiation boundary case.

The analytical solution for a step change boundary condition, as given by Jenson and Jeffreys (23) is:

$$\theta = \sum_{n=0}^{\infty} (-1)^n \left\{ \operatorname{erfc}\left[(x+2n)/2\sqrt{\tau}\right] + \operatorname{erfc}\left[(2n+2-x)/2\sqrt{\tau}\right] \right\} \qquad (1.169)$$

and for the convection boundary condition by Chapman (6):

$$\theta = 2\sum_{n=1}^{\infty} e^{-\delta_n^2 \tau} \frac{\sin \delta_n}{\delta_n + \sin \delta_n \cos \delta_n} \cos\left(\delta_n (1-x)\right) \qquad (1.170)$$

where δ_n are the roots of:

$$\delta_n \tan \delta_n - Bi = 0 \qquad (1.171)$$

SECTION 2

COMPUTATION, RESULTS AND DISCUSSION

A. Introduction

A large amount of effort has been devoted to the theoretical study of the various aspects of equation (1.1) but few workers attack the numerical solution in a practical manner. It is necessary when making a numerical approach to solve the heat conduction equation to decide the required lattice density and also the amount of computer time it is likely will be spent in obtaining this numerical solution.

The information given in the paper by Campbell and Vollenweider (5) does suggest that a practical approach to solving heat conduction problems has been made but it appears that information about their program is unobtainable.

Two methods were selected for the problem of one-dimensional heat conduction in a plate with step change and adiabatic boundary conditions: the Dusinberre explicit and the Crank-Nicolson implicit method. When these had been investigated for cases where analytical solutions exist, other methods were experimented with for the case of a plate with a radiation boundary condition.

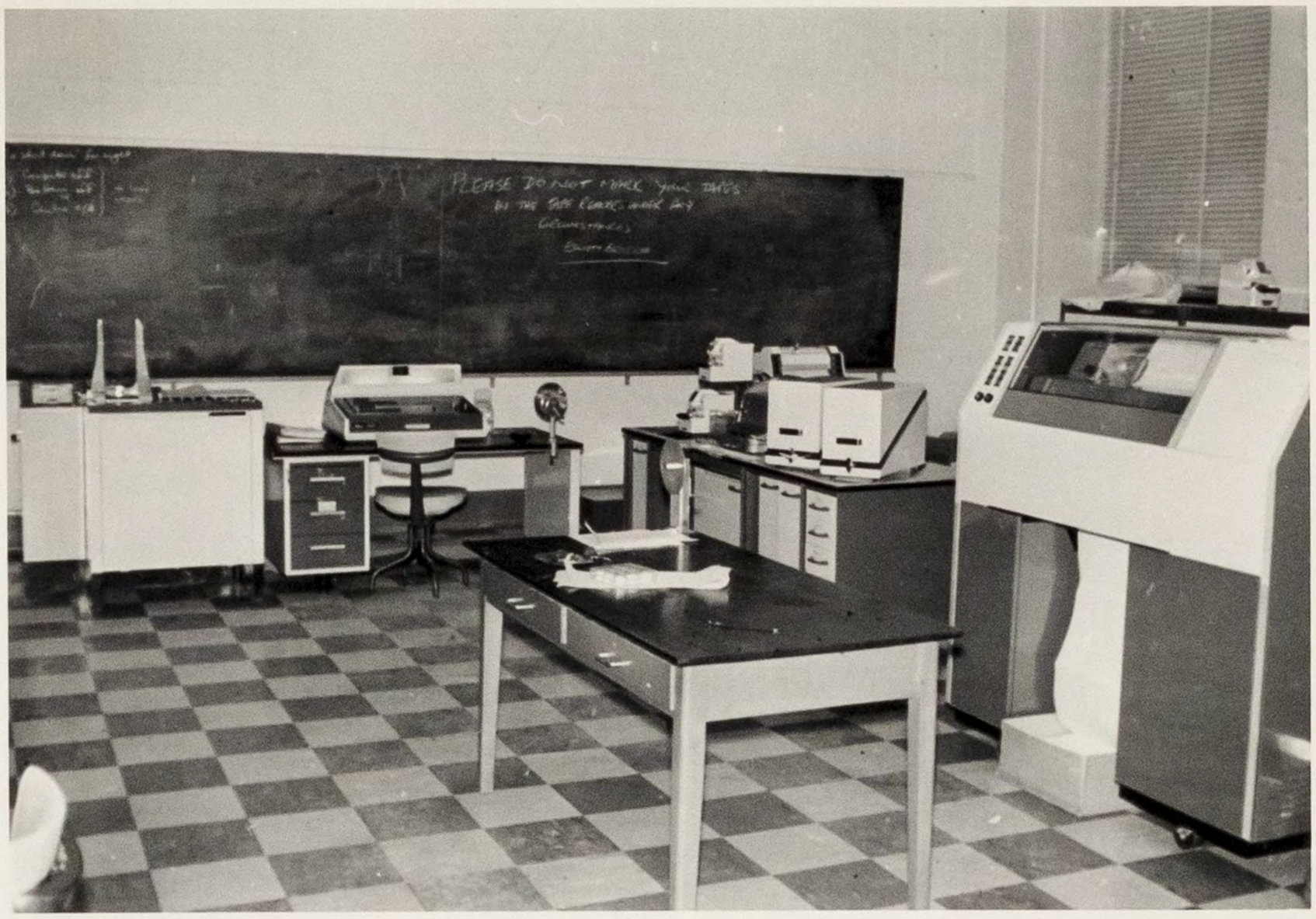
The accuracy of the integral and analogue computer solutions has also been investigated.

The numerical computation was done on the University's Elliott 803B computer. This computer has floating point hardware; 8192 words of core store, each word having 39 bits which can hold one floating point number, one integer

or two instructions; three film handlers; a card reader; a lineprinter and an on-line incremental plotter. Photographs of the installation are shown in fig. 2.1. The Elliott 803B has 576μs fixed point add and 864μs floating point add.

Occasional use was made of the Science Research Council's I.C.T. Atlas at Chilton for some of the longer programs. The floating point add time on this computer is 1.6μs

The analogue computation was performed on an Electronic Associates Hybrid-48 parallel hybrid computer. This machine is basically an analogue computer with digital logic which can be used to control high speed analogue switches and other pieces of analogue equipment. A photograph of the computer is shown in fig. 2.1.


PLEASE DO NOT MARK

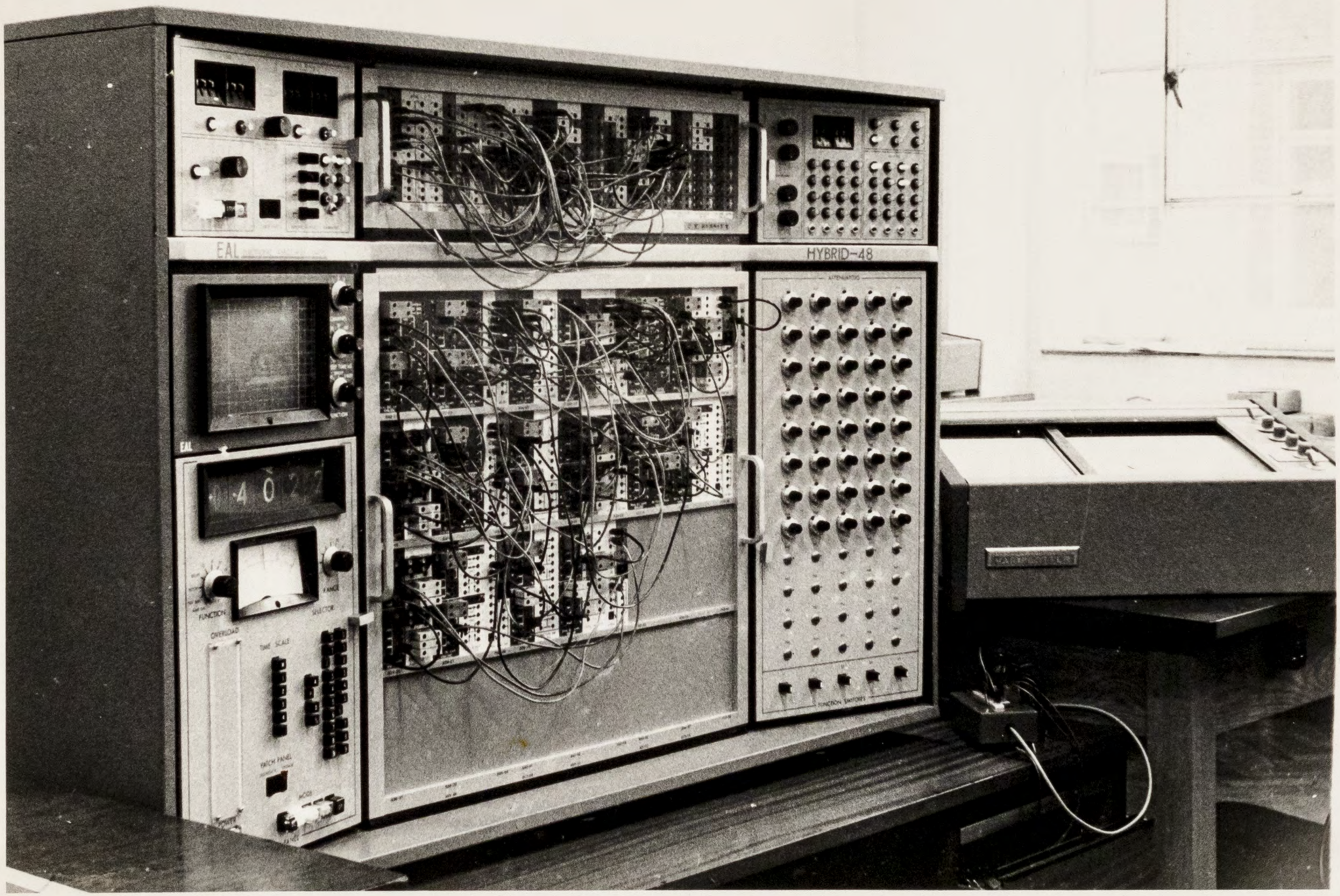

EAL
HYBRID-48
EAL
FUNCTION
RANGE
SELECTOR
OVERLOAD
TIME SCALE
PATCH PANEL
MODE
FUNCTION SWITCHES

FIG 2.1

B. Computation times for the explicit and Crank-Nicolson methods.

Programs were written in Algol to calculate the temperature-time history for a plate with uniform initial temperature distribution, and step change and adiabatic boundary conditions (given by equations (1.24), (1.25) and (1.30)), and run on the Elliott 803. The results are shown in table B.1.

It is seen that the Crank-Nicolson is slower than the explicit method. When 5 increments are used, the explicit is about 50% faster than the Crank-Nicolson method when $M = 0.1$. As M is increased, the speed of the two methods becomes closer. When $N = 10$, the difference in speed is more noticeable; for $M = 0.1$ the explicit is more than twice as fast as the Crank-Nicolson method, and when $M = 0.5$ the explicit is still more than twice as fast as the Crank-Nicolson method.

C. Calculation of the analytical solution for a step change boundary condition.

In order to estimate the accuracy of the two approximate methods it is necessary to know the analytical solution to the problem. The analytical solution used was that given by equation 1.169. θ (X,τ) was calculated for X = 0.2 to 1.0 in steps of 0.2 for values of τ from 0.001 to 0.500 in steps of 0.001. The results were stored on magnetic film. The analytical solutions for the values of X above and selected values of τ are shown in table B.2. These analytical solutions agree with Schneider (37).

D. The behaviour of the explicit method for a high lattice density.

It has been shown mathematically in many works, such as that by Douglas (8), that the explicit solution converges to the analytical solution as the lattice density is refined. It was decided to test this numerically by obtaining an explicit solution and comparing the results with the analytical solution.

The explicit method was used with N = 50 and M = 0.05. Some selected results are shown in table B.3. It is seen that the explicit solution soon diverges from the analytical solution, in fact, the differences are large. This is interesting; the results are probably due to the accumulation of truncation error. Although it has been shown that the explicit method is stable, a large number of time steps may result in accumulation of the errors.

E. Maximum errors in the explicit and Crank-Nicolson methods.

Algol programs were written for the explicit and Crank-Nicolson methods for a plate with adiabatic and step change boundary conditions. The solutions obtained were compared with the analytical solution and the results were printed in the form shown by table B.6.

The tables of results were examined to select the maximum errors for each nodal point for values of M from 0.1 - 0.5 in the case of the explicit and 0.1 - 1.0 for the Crank-Nicolson method. These results are presented in tables B.4 and B.5.

When these errors were selected, it was noticed that for X = 0.2, the maximum error occurred at the lower values of τ, which was to be expected as the temperature of the plate is varying more rapidly. When X = 1, it was noticed that the maximum errors occurred at higher values of τ as the temperature at this boundary is not changing very quickly at lower values of τ.

It has been noticed from the literature that, in general, the results of finite difference approximations to the heat conduction equation have been presented in the form of tables similar to B.6. However, with the availability of on-line incremental plotters, it is now possible to obtain the results graphically; the accuracy of the various methods

can be readily seen by reference to a graph.

The accuracy of the explicit and Crank-Nicolson methods for various values of N and M can be seen from figs. C.4 - C.11. Only a limited number of graphs are presented in order to save space. Graphs showing the accuracy of the explicit and Crank-Nicolson methods were plotted for N = 5, 10, 15 and M = 0.1 to 0.5 in steps of 0.1 for the explicit method and 0.1 to 1.0 in steps of 0.1 for the Crank-Nicolson method. It can be seen that it is much easier to examine the accuracy of the methods from a graph than from a table of figures such as those given in B.6.

Figs. C.4 and C.5 show the accuracy of the explicit method for M = 0.1 and 0.5 respectively when N = 5. With M = 0.1, none of the nodal points exhibit errors > 2%. When M = 0.5 (the limit for stability) several points show errors > 2% (the convention in the graphs is that if a point lies just below the 2% line, the error is greater than, or equal to, 2%). Note that when X = 1, the accuracy is good for the first few values of τ, whereas when X = 0.2, the first few points are inaccurate.

When N is increased to 10, the accuracy is increased and it appears from figs. C.4 and C.7 that N = 5 and M = 0.1 exhibits similar accuracy characteristics to N = 10 and M = 0.5. Reference to table B.1, however, shows that the explicit method with N = 5 and M = 0.1 takes 42.8 secs.

whereas when N = 10 and M = 0.5 it takes 55 secs. computation time.

Fig. C.8 shows the explicit method with N = 15 and M = 0.1. As expected, the accuracy is increased.

Fig. C.9 shows the results obtained using the Crank-Nicolson method with N = 5, M = 0.1 for the same problem. It is noticed that this method is less accurate than the explicit method with the same lattice density. When M is increased to 0.5 (Fig. C.10), it is seen that the error is on the whole larger than that obtained using the explicit method. Fig C.11 shows the Crank-Nicolson method for M = 1.0. The results are inaccurate but the method remains stable.

The results obtained using the Crank-Nicolson method are higher than those obtained for the explicit method. Reference to Fig. 2.2 explains this; it is noticed that the gradient at time τ (that used in the explicit formulation) is greater than that used by the Crank-Nicolson at $\tau + \frac{1}{2}\Delta\tau$ consequently, the nodal temperature at $\tau + \Delta\tau$ calculated by the explicit are lower than those calculated by the Crank-Nicolson method. It is to be expected that the backward difference would produce results which had a higher positive error than the Crank-Nicolson method, thus this method would be less accurate. As in the case of the explicit method, increasing N makes the Crank-Nicolson more accurate.

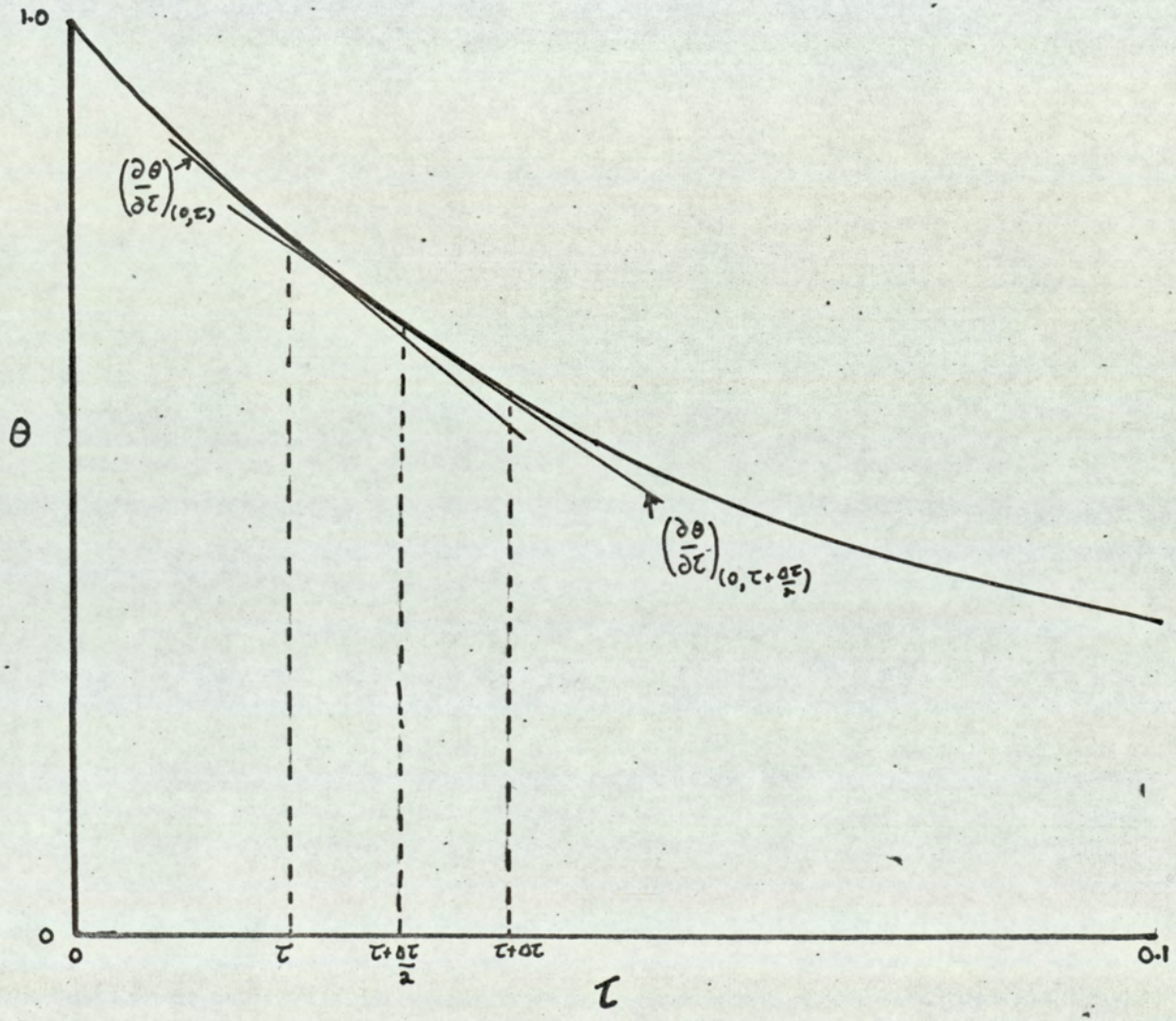


FIG 2.2 GRADIENTS OF θ v τ CURVE AT τ AND $\tau+\frac{1}{2}\Delta\tau$ FOR A PLATE WITH ADIABATIC AND STEP CHANGE BOUNDARY CONDITIONS. X=O

F. Maximum modulus for various boundary approximations.

The natural approximation to use at a boundary where convection or radiation is the mode of heat transfer is that given by equation (1.33). This was used in an analogue computer for the convection boundary problem, Bi = 10 and it was noticed that the results were inaccurate. The approximation given by equation (1.32) also gave inaccurate results. The results were, however, inaccurate in the other direction, consequently equations (1.32) and (1.33) were averaged to give.

$$\left(\frac{\partial\theta}{\partial X}\right)_{(0,\tau)} = \frac{1}{2}\left[\frac{\theta_{(0,\tau)} - \theta_{(-1,\tau)}}{\Delta X} + \frac{\theta_{(1,\tau)} - \theta_{(-1,\tau)}}{2\Delta X}\right] \qquad (2.1)$$

This approximation gave accurate results.

Applying Dusinberres stability analysis given by equation (1.81) showed that higher values of M could be used (i.e. the method remained stable) with the boundary approximation given by equation (1.32) than with the approximation given by equation (1.33). Fig. C.12 presents graphically the maximum permissible value of M as a function of Bi/N for the explicit method when any of the three boundary approximations (equations (1.32), (1.33), (2.1)) are used.

G. The analytical solution for the convection boundary problem.

It was decided to use a similar approach to estimate the accuracy of the explicit and Crank-Nicolson methods when there is convection at the boundary X = 0, as was used in the case of a step change boundary condition. The analytical solutions for Bi = 0.1, 1.0 and 10.0 were calculated for values of X from 0 to 1.0 in steps of 0.2 and τ from 0.001 to 1.000 in steps of 0.001 using equation (1.170). The results were stored on magnetic film. Table B.7 shows selected results.

H. Maximum errors for the convective boundary problem

Programs were written in Algol to compute the temperatures at various values of τ for a plate with convective and adiabatic boundary conditions and uniform initial temperature distribution. As with the step change boundary, temperatures were only output for the selected values of $\tau \geq$ 0.04, 0.08, 0.12 ... 1.00. When there is convection at X = 0, $\theta_{(o,\tau)}$ varies with τ, consequently values of $\theta_{(o,\tau)}$ were recorded. The form of the results was similar to table B.6. As before, tables of numerical data are tedious to examine, therefore the results were graphed in a similar manner to those obtained for the step change problem. Selected graphs are presented as figs. C.13 - C.29. The three boundary approximations mentioned above were examined and some of the graphs demonstrate the accuracies for the boundary approximations other than that given by equation (1.33). The explicit method was tested for N = 5, 10, 15 with M = 0.1, 0.2 ... 0.5. The Crank-Nicolson method was tested for the same values of N but M = 0.2, 0.4 ... 1.4.

Tables B.8 - B.18 were constructed from the numerical results and these give a good indication of the performance of the two methods. Only the boundary at X = 0 is considered in these tables in order to save space.

The results shown in table B.8 compare the accuracies

of the explicit and Crank-Nicolson methods for Bi = 0.1. It is seen that both the methods are accurate and in both cases the maximum error occurs at the lower values of τ. The advantage of the superior stability of the Crank-Nicolson method is demonstrated; this method can be used for values of M higher than 1.4 as at this value, the maximum error is only 0.22%.

Table B.9 shows the two methods for Bi = 1.0. It is noticed that the faster rate of cooling of the surface temperature has a detrimental effect on the accuracy. It is also seen that the explicit method becomes unstable for M = 0.5; this fact is borne out by fig. C.12, since for Bi/N = 0.2 it is seen that $M \leq 0.35$. However, the explicit method does remain stable for $M \leq 0.4$. When M = 0.4, the maximum error is only 1.5%. The stability of the Crank-Nicolson for $M > 0.5$ is still advantageous. When M = 1.0, the maximum error is 2%. For M = 1.2 and 1.4 only one point in each case has error > 2%. It appears that for smaller values of M, the Crank-Nicolson method is less accurate than the explicit method but at higher values of M, the accuracy is superior to the explicit method.

Table B.10 shows the performance of the two methods for Bi = 10. The explicit method is unstable for $M \geq 0.3$. The Crank-Nicolson method appears to be good for $M > 0.4$; only 8% of the points have errors > 2%.

Table B.11 demonstrates the advantage of increasing N. The explicit method is fairly accurate for $M \leq 0.3$. The superior accuracy of the Crank-Nicolson method is shown as it is good for $M \leq 1.2$. Selected graphs for the two methods for the boundary condition given by equation (1.32) are shown as figs. C.13 - C.23. No figures are presented for Bi = 0.1 as accurate results are achieved for most values of M (see table B.8). Figs. C.13 and C.14 show the explicit method for Bi = 1.0. It is seen that the explicit method is accurate for M = 0.1 but inaccurate for M = 0.5 (compare figs. C.4 and C.5 for the step change boundary condition). Figs. C.15 and C.16 show the effect of increasing N from 5 to 10 for Bi = 10.0. Note the increase in accuracy for all nodal points. Fig. C.17 shows the effect of increasing M when N = 10. It is seen that inaccurate results are obtained at lower values of τ but the accuracy improves as increases. Fig C.18 shows the effect of a further increase in N. Fig. C.19 proves the Crank-Nicolson method to be accurate for Bi = 1.0 and M = 0.2. When M is increased to 1.4, the Crank-Nicolson method is accurate as shown by fig. C.20 at higher values of τ but at the lower values the solution is oscillatory. Gay (19) and Greenwood (21) have observed this behaviour. Fig. C.21 shows the Crank-Nicolson method for Bi = 10. Comparison with fig. C.15 shows the Crank-Nicolson to be more accurate. It appears as though

the inherent nature of the Crank-Nicolson method to produce higher results than the explicit method has to some extent been cancelled out by the error produced by the approximation at the boundary.

It is seen from fig. C.22 that the Crank-Nicolson method produces accurate results at higher values of τ for Bi = 10.0 when N = 10 and M = 1.4. The same oscillatory nature is realised as in the case of Bi = 1.0. Fig. C.23 shows the Crank-Nicolson method with N = 10 and M = 0.2.

Tables B.13 - B.15 illustrate the explicit and Crank-Nicolson methods for the boundary approximation given by equation (1.32). It is seen that this approximation is unsatisfactory; even with Bi = 0.1 there is a maximum error of 1.4% for both the explicit and Crank-Nicolson methods. This error is independent of M and it can be seen that it occurs at the highest value of τ. For Bi = 1.0 and 10.0 the errors are increased proportionately. It is interesting to note that for Bi = 1.0, 96% of the points exhibit errors >2% for the explicit method but for Bi = 10 most of the points have errors of >2% only when M = 0.1. As M is increased, the number of points with errors > 2% decreases. The explicit method remains stable for $M \leq 0.4$. With the Crank-Nicolson method, 96% of the points have errors > 2% for all values of M.

Graphs representing this boundary condition are shown

as figs. C.24 and C.25. Fig C.24 demonstrates the error increasing as τ increases for Bi = 0.1. When Bi = 10.0, the maximum error does not occur at the high values of τ. The error tends towards a constant value.

Tables B.16 - B.18 show the results obtained with the boundary approximation given by equation (2.1). For Bi = 0.1, the accuracy is better than that obtained by using equation (1.32). As before, the maximum error is independent of M and occurs at the highest value of τ for both explicit and Crank-Nicolson methods. When Bi = 1.0, all the points exhibit errors greater than 2% for both methods. When Bi = 10.0, this approximation at the boundary appears to be better than that given by equation (1.33) when the explicit method is used and only slightly better when the Crank-Nicolson method is used. Figs. C.26 - C.29 illustrate this boundary approximation. Figs. C.26 and C.27 show the approximation to be poor for Bi = 0.1 and 1.0. When Bi = 10, the explicit method is surprisingly accurate (compare with figs. C.25 and C.15). This approximation does not have the same beneficial effect on the Crank-Nicolson method.

J. An "exact" solution for the radiation boundary problem

After examination of the results for the step change and convective boundary conditions, it was decided that the results using the explicit method with N = 15 and M = 0.1 were accurate for Bi = 10, consequently it was decided to use this method to evaluate the temperature-time history of a plate with radiation and adiabatic boundary conditions for R = 10. The results obtained using this method for the convective boundary problem, together with the accuracy of each point for the surfaces X = 0 and X = 1 are shown in table B.19. Table B.20 shows the solutions obtained for the radiation boundary condition. It was decided only to pay attention to the case for R = 10.0 because in the convective boundary case, the method for Bi = 10 produced less accurate results than either Bi = 0.1 or Bi = 1.0.

Although the Crank-Nicolson method gave results which were slightly more accurate than the explicit, it was decided to use the explicit method because the solutions could be obtained directly without iteration (iteration is necessary with the implicit methods because of the non-linearity at the radiating boundary). Consequently the accuracy of the solution will not depend on the chosen iteration technique.

K. Behaviour of the implicit methods in the radiation boundary problem.

It was decided at this stage to introduce the use of the backward difference method. (See equation 1.20).

In order to use the implicit methods, it was necessary to choose an iteration algorithm; this was taken as equation (1.103). Algol programs were written for both the Crank-Nicolson and backward difference methods to determine the required number of iterations to establish a convergence of 0.00001. The results for R = 10, N = 5 and M = 0.2 are given in table B.21. It is noticed that the backward difference requires more iterations than does the Crank-Nicolson method. This removes the advantage of less calculation to evaluate the right hand side of the backward difference formulation than is required for the Crank-Nicolson.

L. Efficiency of the numerical methods

The explicit method using N = 15, M = 0.1 and R = 10 is accurate but time consuming. It was decided to compare the results of five techniques using N = 5 and various values of M with the explicit "exact" solution and to measure the computation times on the Elliott 803 B computer for the solution to reach $\tau \gtrsim 0.95$. The five techniques are: explicit, Crank-Nicolson using simple iteration, backward difference using simple iteration (see equation (1.103)), Crank-Nicolson using Newton-Raphson iteration and the backward difference method using Newton-Raphson iteration. (See equation (1.104).

The computer programs were written in Elliott 803 machine code; this was chosen in order to eliminate any effect of compiler structure on the computation time. Programs for the explicit method and the backward difference method using both simple and Newton-Raphson iteration are shown in appendix A.

The results are shown in table B.22 and graphically in figs. C.30 - C.33.

Fig. C.30 shows the computation time as a function of M. It is seen that the explicit method is the fastest. For low values of M, the backward difference method using Newton-Raphson iteration is slowest with the Crank-Nicolson method

almost as slow.

The backward difference method using simple iteration exhibits interesting behaviour; the computation time decreases as M is increased up to a certain value and further increase in M slows the method down until when $M > 0.8$ the algorithm does not converge. This can be explained:

Considering the approximation at the boundary X = 0

$$-2M\theta_{(1,\tau+\Delta\tau)} + (1+2M)\theta_{(0,\tau+\Delta\tau)} + 2M\Delta X R\theta^4_{(0,\tau+\Delta\tau)} = \theta_{(0,\tau)} \qquad (2.2)$$

The initial estimate for $\theta_{(0,\tau+\Delta\tau)}$ is $\theta_{(0,\tau)}$

Using the method of solution for the tri-diagonal matrix given in Modern Computing Methods, the iterate for $\theta_{(0,\tau+\Delta\tau)}$ is given:

$$\theta^{(m+1)}_{(0,\tau+\Delta\tau)} = (\delta_0 - c_0\,\theta_{(1,\tau+\Delta\tau)})/\beta_0 \qquad (2.3)$$

where $\beta_0 = 1+2M + 2M\Delta X R\,\theta^{(m)\,3}_{(0,\tau+\Delta\tau)}$

$$\delta_0 = \theta_{(0,\tau)}$$

$$\therefore\ \theta^{(m+1)}_{(0,\tau+\Delta\tau)} = \frac{\theta_{(0,\tau)} + 2M\theta_{(1,\tau+\Delta\tau)}}{(1+2M) + 2MR\Delta X\,\theta^{(m)\,3}_{(0,\tau+\Delta\tau)}} \qquad (2.4)$$

for the first iteration cycle, $\theta_{(0,\tau)} = 1$

$$\therefore\ \theta^{(m+1)}_{(0,\tau+\Delta\tau)} = \frac{1 + 2M\theta_{(1,\tau+\Delta\tau)}}{(1+2M) + 2MR\Delta X\,\theta^{(m)\,3}_{(0,\tau+\Delta\tau)}} = f(\theta_{(0,\tau+\Delta\tau)}, \theta_{(1,\tau+\Delta\tau)}) \qquad (2.5)$$

For various selected values of $\theta_{(0,\tau+\Delta\tau)}$ using equation (2.5) in conjunction with the elimination method given in Modern Computing Methods, fig. 2.3 was drawn for (a) M = 0.2 and (b) M = 1.0. The simple iteration procedure is represented on this figure. It is readily seen that for M = 0.2, the iteration algorithm converges but for M = 1.0 it is impossible to achieve convergence.

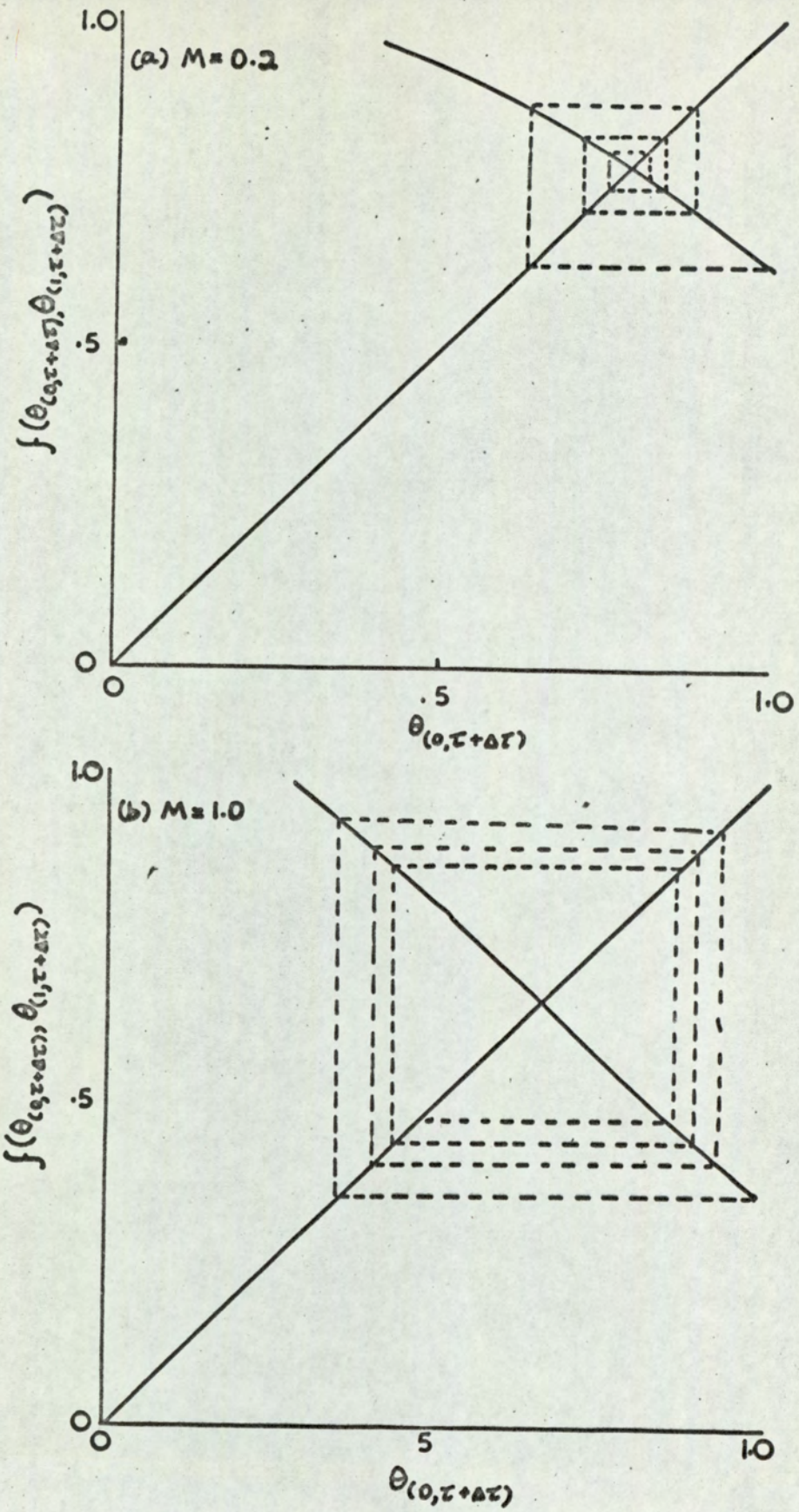


FIG 2.3 SIMPLE ITERATION SOLUTION

When a smaller value of R is considered (e.g. 1.0) the algorithm does converge for $M > 0.8$.

The Crank-Nicolson method behaves in a similar manner but for R = 10 it is possible to use values of $M \leq 1.2$.

Fig. C.31 shows the maximum error in $\theta_{(0,\tau)}$ as a function of M (The peculiar shapes of the curves are unfortunate but cannot be avoided because values of $\theta_{(n,\tau)}$ (n = 0 or n = N) are output only when $\tau \geq 0.04, 0.08 \ldots 1.0$). It is seen that there is a rapid increase in the error when the explicit method is used. The points for M = 0.35 and 0.40 lie outside the range of the graph (i.e. $E > 10\%$). As expected, the Crank-Nicolson method using the two iterative techniques lie on the same curve as do those for the backward difference method. It is interesting to note that the backward difference method using Newton-Raphson iteration reaches a maximum error at M = 1.0 and further increase in M fails to increase this error. The backward difference method using Newton-Raphson iteration converges for all values of $M \leq 2.0$. The method was not tested for $M > 2.0$. Fig C.31 shows the Crank-Nicolson method to become inaccurate at higher values of M.

Reference to table B.22 shows the backward difference method to be most accurate. The Crank-Nicolson method with M = 1.0 exhibits a maximum error of 74.3% when simple iteration is used. The backward difference method for M = 1.0 exhibits a maximum error of only 9.3%.

Fig. C.32, which illustrates the maximum errors at the insulated boundary, shows the superior accuracy of the backward difference method. Again, with the explicit method, there is a rapid increase in the error as M is increased.

Fig. C.33 shows the average error at the radiating boundary. It is interesting to note that the average error for the backward difference method is low.

M. Iterative methods for solving the simultaneous equations

The iterative methods of solving sets of simultaneous equations are described by equations (1.105) - (1.111).

Iterative methods of solution are more advantageous in two and three-dimensional problems whether linear or non-linear. It seemed a logical step to try and improve on the computation time for the backward difference method with a radiation boundary condition. Since the problem is non-linear, it lends itself to iterative methods. The Jacobian and Gauss-Seidel methods of iteration were compared by measuring the number of iteration cycles required to establish convergence for each method using various values of M and R. The results are shown in tables B.23 - B.25.

Table B.23 shows there is little difference in the relative merits of the two methods. In both cases, the number of iteration cycles required to establish convergence is small. Table B.24 shows the superiority of the Gauss-Seidel method. For M = 0.1, the number of iterations is almost the same for both methods. As M is increased, with the Jacobian method, the number of iterations progressively increases. The behaviour is not so marked with the Gauss-Seidel method. Table B.25 shows the results for R = 10. The differences between the two methods is amplified here. Note that the method has been tried for various values of

$M \leq 2.0$. The Jacobian scheme fails to converge when $M > 0.8$ (compare the simple iteration in part M). It is seen from table B.25 that the required iterations decreases as M is increased beyond a certain value when the Gauss—Seidel method is used.

Table B.26 illustrates the relaxation method; it appears that when $\omega = 0.7$, the least number of iteration cycles to establish convergence are required. On examining these results it was decided to compare the efficiency of the back-ware difference method using relaxation with that when Newton-Raphson iteration is used. The results are shown in figs. C.34 - C.36.

Fig. C.34 shows the saving in computing time when the relaxation method was used; both methods were iterated to a convergence of 0.005. When $M = 0.1$, the relaxation method is much slower than $M \geq 0.2$. It is noticed that as M is increased, the computation times become closes together.

Figs. C.35 and C.36 prove the relaxation method to be disappointing and illustrate the danger in relying on the difference between two successive iterates being close to one another to confirm that the solution has converged. Theoretically, when both the relaxation and Newton-Raphson methods have reached the desired convergence, they should have the same accuracy because both methods are solving the

TABLE 2.1

PROGRESS OF RELAXATION SOLUTION
FOR THE BACKWARD DIFFERENCE METHOD.

ITERATION NO.	NODE					
	0	1	2	3	4	5
	1.0	1.0	1.0	1.0	1.0	1.0
1	0.6000	0.8667	0.9556	0.9852	0.9951	0.9968
2	0.6752	0.8814	0.9570	0.9845	0.9940	0.9961
3	0.6598	0.8734	0.9531	0.9807	0.9924	0.9951
4	0.6613	0.8723	0.9515	0.9810	0.9922	0.9948

same sets of equations; this fact is demonstrated by figs. C.31 and C.32. Comparison of figs. C.35 and C.36 show that the relaxation solution is less accurate at the insulated boundary than at the radiating boundary; usually the reverse is true. This can be simply explained. Table 2.1 shows the progress of the relaxation solution with $\omega = 0.7$ for the radiation boundary problem for the first time step. It is noticed that at the boundary X = 0, the iterates oscillate about the converged solution. At the boundary X = 1, the iterates approach the converged solution without oscillation. This results in a positive error. This error will accumulate with errors produced at future time steps.

One further point about the convergence of relaxation methods can be noticed from equation (1.111): the difference between successive iterates is given by $\omega \left[\bar{\theta}^{(m+1)}_{(n,\tau+\Delta\tau)} - \theta^{(m)}_{(n,\tau+\Delta\tau)} \right]$ therefore if a convergence of 0.005 is set, iteration is really continued until a convergence of $0.005/\omega$.

N. Solution of two and three dimensional problems

The one-dimensional heat conduction equation is applicable in only a limited number of cases. A larger number of problems can be solved by assuming two-dimensional heat flow. The solution of a two-dimensional equation by a difference method requires much more computation; therefore in this case, the relative efficiencies of the methods are bound to be more noticeable. Bearing these facts in mind, it was decided to test the efficacy of three different two-dimensional methods: the explicit, alternating direction implicit and alternating direction explicit methods. Allada and Quon present some results on the efficiencies of the alternating direction implicit and explicit methods for a three-dimensional problem; the methods are shown to be inaccurate.

The two-dimensional problem considered was that of a long square bar, initially at a uniform temperature throughout with step changes in temperature at its surfaces when $t = 0$. Represented mathematically this is:-

$$\frac{\partial \theta}{\partial \tau} = \frac{\partial^2 \theta}{\partial X^2} + \frac{\partial^2 \theta}{\partial Y^2} \qquad (2.6)$$

assuming the physical properties are isotropic and independent of temperature and, Lx = Ly. The initial and boundary conditions are:

$$\tau < 0 \quad 0 \leq X \leq 1 \quad 0 \leq Y \leq 1 \quad \theta(X, Y, 0) = 1 \qquad (2.7)$$

$$\tau \geq 0 \quad X=0 \text{ and/or } Y=0 \qquad \theta(x, Y, \tau) = 0 \tag{2.8}$$

$$Y=1 \qquad \frac{\partial \theta}{\partial Y}(x, 1, \tau) = 0 \tag{2.9}$$

$$X=1 \qquad \frac{\partial \theta}{\partial X}(1, Y, \tau) = 0 \tag{2.10}$$

The methods used to obtain a solution are expressed as equations (1.48), (1.53), (1.54) and the alternating direction explicit method as:

$$\theta_{(m,k,\tau+\Delta\tau)} = \frac{\theta_{(m,k,\tau)}}{(1+2M)} + \frac{M}{(1+2m)}\left[\theta_{(m-1,k,\tau+\Delta\tau)} - 2\theta_{(m,k,\tau)} + \theta_{(m+1,k,\tau)}\right.$$

$$\left. + \theta_{(m,k-1,\tau+\Delta\tau)} + \theta_{(m,k+1,\tau)}\right] \tag{2.11}$$

The analytical solution for X = Y = 0.2 is obtained by squaring the values given for X = 0.2 in table 1.2.

The results are shown in tables B.27 - B.29.

Table B.27 shows the explicit methods to be more than twice as fast as the alternating direction implicit method. Table B.28 shows the alternating direction implicit method to be slightly less accurate than the explicit method. Table B.29 shows the alternating direction explicit method to be inaccurate.

The implicit methods with radiation and adiabatic boundary conditions.

The explicit method can be recommended for two-dimensional problems but one may desire to use an implicit method for a radiation boundary problem when the temperatures

are varying slowly. The alternating direction implicit method is to be recommended since the matrix of coefficients is tri-diagonal. To obtain a solution for the case of a long square bar with radiation and adiabatic boundary conditions would require the solution of N + 1 simultaneous (tri-diagonal) equations N + 1 times. The method of solution is:

Fig. 2.4 shows the cross-section of the bar divided into 5 increments. The difference equations are explicit in the k direction and implicit in the n direction for one time step then vice-versa for the next.

The simultaneous equations for any k are:

$$-(2+S)\theta_{(0,k,\tau+\Delta\tau)}+2\theta_{(1,k,\tau+\Delta\tau)} = d_0 \quad (2.12)$$

$$\theta_{(0,k,\tau+\Delta\tau)}-(2+S)\theta_{(1,k,\tau+\Delta\tau)}+\theta_{(2,k,\tau+\Delta\tau)} = d_1$$

$$\vdots$$

$$\theta_{(3,k,\tau+\Delta\tau)}-(2+S)\theta_{(4,k,\tau+\Delta\tau)}+\theta_{(5,k,\tau+\Delta\tau)} = d_4$$

$$\theta_{(4,k,\tau+\Delta\tau)}-\left[(2+S)-2R\Delta\times\theta^{3}_{(5,k,\tau+\Delta\tau)}\right]\theta_{(5,k,\tau+\Delta\tau)} = d_5$$

where d_0, d_1, ... d_5 are obtained from the explicit formulations in the k direction. $\theta_{(n,k,\tau+\Delta\tau)}$ is eliminated in turn from each equation (2.12) using the method given in Modern Computing methods until one non-linear equation involving $\theta_{(5,k,\tau+\Delta\tau)}$ is obtained. This is solved iteratively (e.g. using the Newton-Raphson method). Back substitution

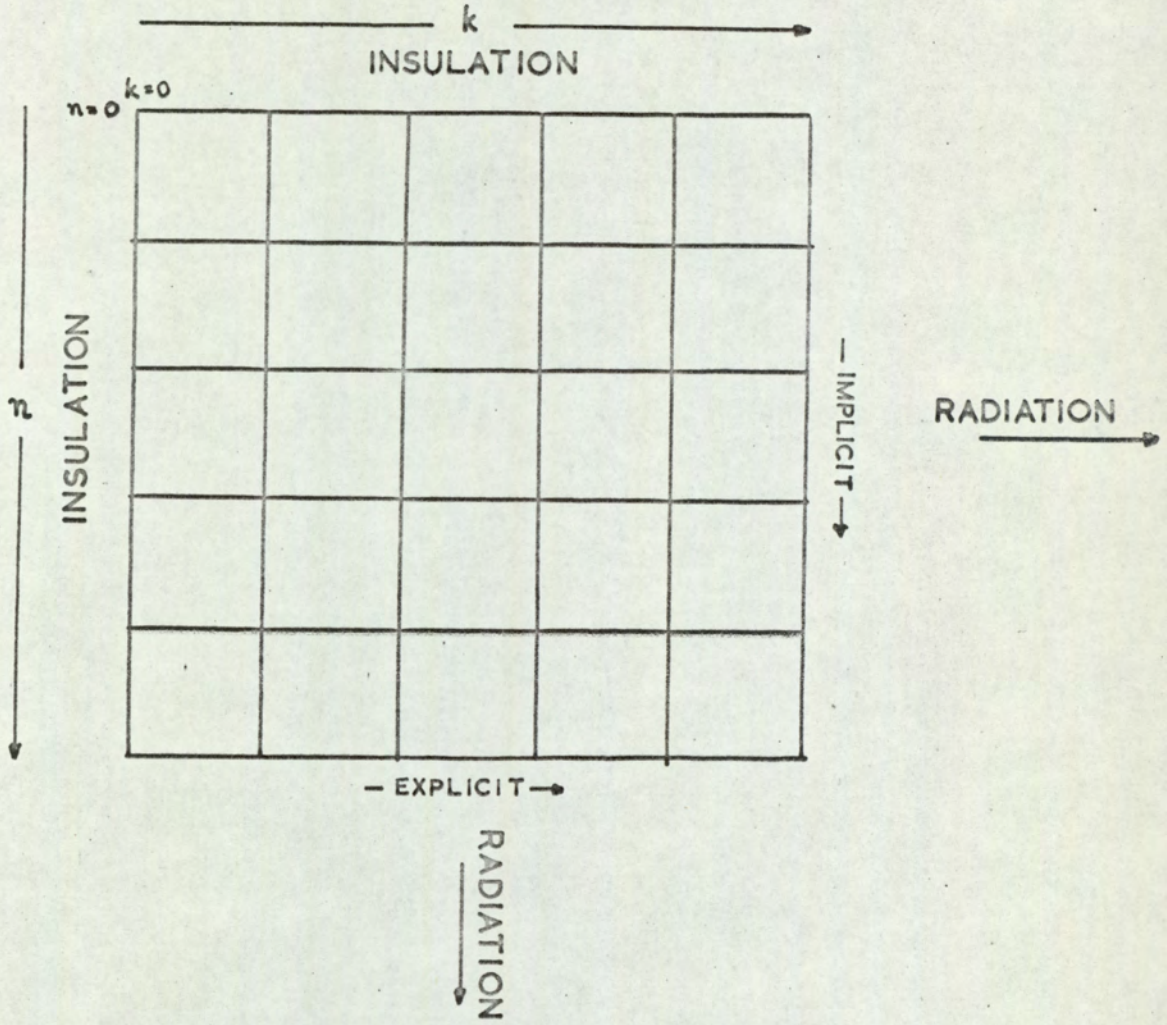


FIG 2·4 ALTERNATING DIRECTION IMPLICIT METHOD

then gives values for the other nodal temperatures. This procedure is repeated for all values of k. It is to be expected that the solution of a two-dimensional problem by this method would take about 6 times longer than the one-dimensional case.

The fully implicit (backward difference) method given by equation (1.50) can also be used to obtain a solution to the problem. The nodes, normally denoted by the subscripts n, k, are numbered as shown in fig. 2.5. The simultaneous equations for each of these nodes are expressed in matrix form in fig. 2.6. The matrix is diagonal but has 5 non-zero elements in each row, therefore, the equations cannot be simply solved as in the tri-diagonal case.

The equations can be expressed:

$$A\,\theta_{(n,\tau+\Delta\tau)} = \theta_{(n,\tau)} \tag{2.13}$$

$$\therefore\ \theta_{(n,\tau+\Delta\tau)} = A^{-1}\,\theta_{(n,\tau)} = B\theta_{(n,\tau)} \tag{2.14}$$

11 of the equations are non-linear thus an iterative technique has to be used to obtain a solution. One possibility would be to estimate values of $\theta_{(n,\tau+\Delta\tau)}$ at the non-linear nodes, invert the matrix and calculate new values of $\theta_{(n,\tau+\Delta\tau)}$ using equation (2.14). Iteration could be continued until convergence is achieved. The inversion of a 36 x 36 matrix is, however, time consuming. To speed the solution, the following method is proposed:

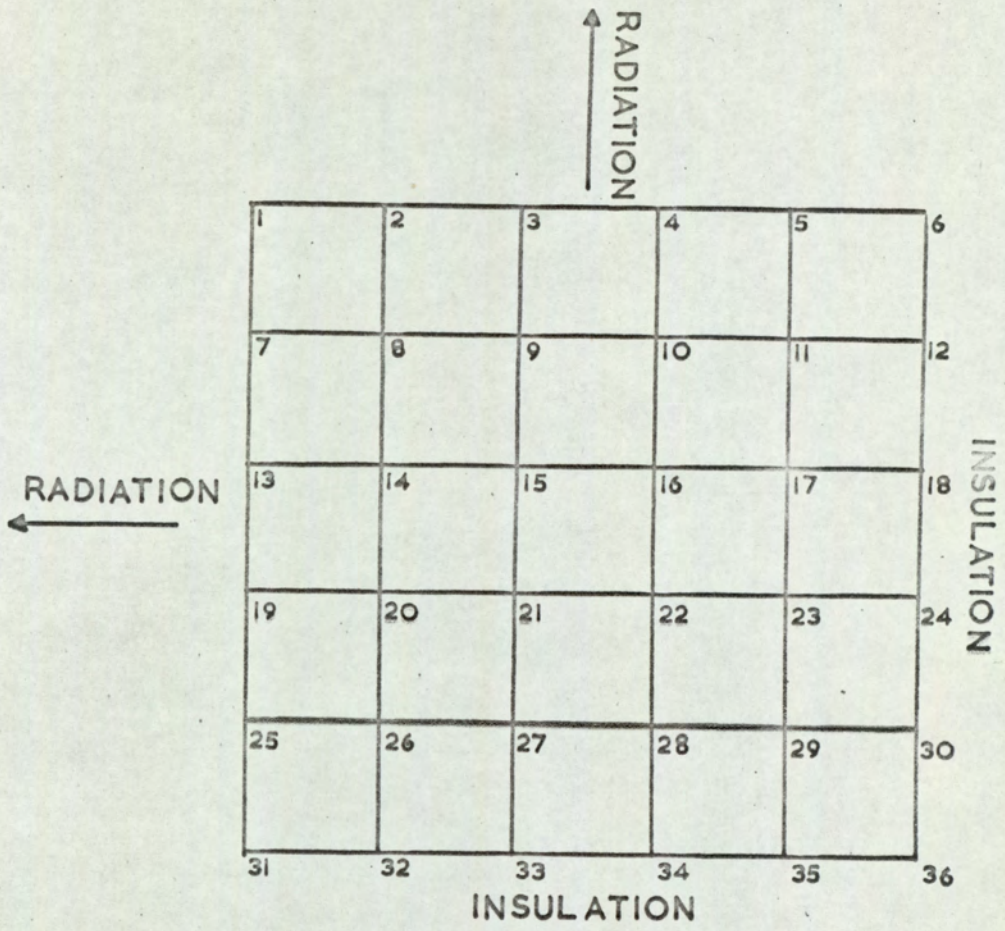


FIG 2.5 IMPLICIT METHOD

$K = 2.M.R.\Delta X$

FIG 2.6

The equations are rearranged as shown in fig. 2.7 and the matrix A is partitioned to form 4 sub-matrices A_{11} , A_{12} , A_{21} , A_{22} . All the non-linear elements appear in A_{22}. The inverse, B, of A is given:

$$B_{11} = A_{11}^{-1} + (A_{11}^{-1} A_{12}) S^{-1} (A_{21} A_{11}^{-1}) \qquad (2.15)$$

$$B_{12} = -(A_{11}^{-1} A_{12}) S^{-1} \qquad (2.16)$$

$$B_{21} = -S^{-1} (A_{21} A_{11}^{-1}) \qquad (2.17)$$

$$B_{22} = S^{-1} \qquad (2.18)$$

and $$S = A_{22} - A_{21} (A_{11}^{-1} A_{12}) \qquad (2.19)$$

A_{11} is inverted initially (once A_{11}^{-1} has been calculated it need not be recalculated for succeeding time steps provided M remains constant). The calculation of the terms $(A_{11}^{-1} A_{12})$, $(A_{21} A_{11}^{-1})$ and $A_{21}(A_{11}^{-1} A_{12})$ is not time consuming since A_{12} and A_{21} are very sparse. The iteration cycle now consists of estimating values of $\theta_{(n,\tau+\Delta\tau)}$ in A_{22}, calculating S, inverting S, calculating B_{21} and finally a closer approximation of $\theta_{(n,\tau+\Delta\tau)}$. Note that during the iterating cycle, it is only necessary to calculate B_{21} and B_{22} in order to obtain the new estimates. When convergence has been achieved, B_{11} and B_{12} can also be calculated, hence $\theta_{(n,\tau+\Delta\tau)}$ for all values of n obtained.

This method was programmed in Algol for the Elliott 803 B computer using N = 5, R = 10 (a high value) and M = 0.5. Only 3 iterations were required to establish convergence,

1+4M	-2M				-2M																															θ_{36}		θ_{36}
-M	1+4M	-M				-2M																														θ_{35}		θ_{35}
	-M	1+4M	-M				-2M																													θ_{34}		θ_{34}
		-M	1+4M	-M				-2M																												θ_{33}		θ_{33}
			-M	1+4M					-2M																	-M											θ_{32}		θ_{32}
-M					1+4M	-2M				-M																										θ_{30}		θ_{30}
	-M				-M	1+4M	-M				-M																									θ_{29}		θ_{29}
		-M				-M	1+4M	-M				-M																								θ_{28}		θ_{28}
			-M				-M	1+4M	-M				-M																							θ_{27}		θ_{27}
				-M				-M	1+4M					-M												-M										θ_{26}		θ_{26}
					-M					1+4M	-2M				-M																					θ_{24}		θ_{24}
						-M				-M	1+4M	-M				-M																				θ_{23}		θ_{23}
							-M				-M	1+4M	-M				-M																			θ_{22}		θ_{22}
								-M				-M	1+4M	-M				-M																		θ_{21}		θ_{21}
									-M				-M	1+4M					-M								-M									θ_{20}		θ_{20}
										-M					1+4M	-2M				-M																θ_{18}		θ_{18}
											-M				-M	1+4M	-M				-M															θ_{17}		θ_{17}
												-M				-M	1+4M	-M				-M														θ_{16}		θ_{16}
													-M				-M	1+4M	-M				-M													θ_{15}	=	θ_{15}
														-M				-M	1+4M					-M				-M								θ_{14}		θ_{14}
															-M					1+4M	-2M									-M						θ_{12}		θ_{12}
																-M				-M	1+4M	-M									-M					θ_{11}		θ_{11}
																	-M				-M	1+4M	-M									-M				θ_{10}		θ_{10}
																		-M				-M	1+4M	-M									-M			θ_{9}		θ_{9}
																			-M				-M	1+4M										-M		θ_{8}		θ_{8}
				-2M																					1+4M $+K\theta_{31}^3$	-2M										θ_{31}		θ_{31}
									-2M																-M	1+4M $+K\theta_{25}^3$	-M									θ_{25}		θ_{25}
														-2M												-M	1+4M $+K\theta_{19}^3$	-M								θ_{19}		θ_{19}
																			-2M								-M	1+4M $+K\theta_{13}^3$	-M							θ_{13}		θ_{13}
																												-M	1+4M $+K\theta_{7}^3$						-M	θ_{7}		θ_{7}
																				-2M										1+4M $+K\theta_{6}^3$	-2M					θ_{6}		θ_{6}
																					-2M									-M	1+4M $+K\theta_{5}^3$	-M				θ_{5}		θ_{5}
																						-2M									-M	1+4M $+K\theta_{4}^3$	-M			θ_{4}		θ_{4}
																							-2M									-M	1+4M $+K\theta_{3}^3$	-M		θ_{3}		θ_{3}
																								-2M									-M	1+4M $+K\theta_{2}^3$	-M	θ_{2}		θ_{2}
																													-2M					-2M	1+4M $+2K\theta_{1}^3$	θ_{1}		θ_{1}

$\tau+\Delta\tau$

FIG 2.7

each iteration taking 1 min 50 secs. The inversion of the 11 x 11 matrix (S) took only 25 secs. The inversion of a 36 x 36 matrix would take 15 min 40 secs., but it is recognised that it would be inefficient to use a standard inversion routine for the matrix in fig. 2.6. However, the effect of partitioning the matrix is beneficial. 10 minutes were required to calculate $\theta_{(m,\tau+\Delta\tau)}$ for the first time step. The calculation for successive time steps is quicker since fewer iteration cycles are required, (see table B.21). It is seen that this method is naturally much slower than the alternating direction implicit but with the availability of faster computers this is of little consequence. On the basis of the experience obtained in the one-dimensional case it is to be expected that the implicit method would be more accurate than the alternating direction implicit method.

Allada and Quon have shown the alternating direction implicit method to be unstable in three dimensions. However, the two-dimensional implicit method can be applied to three-dimensional problems by alternating the implicit and explicit directions just as the one-dimensional implicit method has been applied to two-dimensional problems in the form of the alternating direction implicit method. The resulting method is stable.

This alternating direction implicit method was programmed in Algol and run on the Elliott 803 computer for

values of Bi = 1.0 and M = 1.0. It was noticed that quick convergence was obtained; only 3 iteration cycles being necessary. Thus the solution for a three-dimensional problem takes only 6 times longer than a corresponding two-dimensional case.

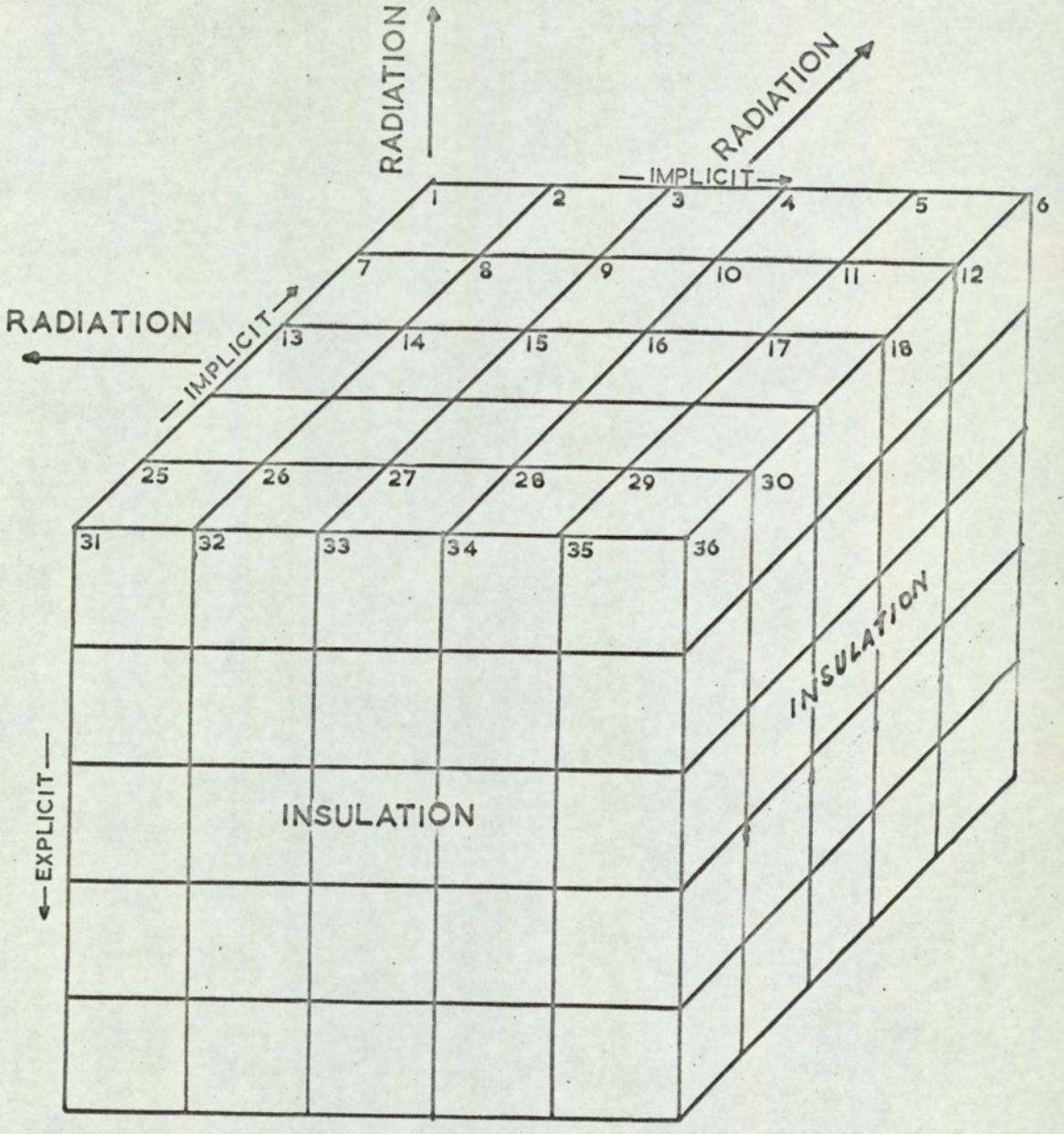


FIG 2.8 THREE-DIMENSIONSIONAL SOLUTION BY THE TWO-DIMENSIONAL IMPLICIT METHOD.

P. Variable Time Steps

Equation (1.66) shows the possibility of using a coarser lattice density when the values of the partial derivatives are small; this occurs when the temperature of the plate is varying slowly.

The results presented by Kardas (24) are a good starting point although they appear to be in error. Table B.30 shows the results computed for the truncation error, at the boundary X = 0 for a plate with convective and adiabatic boundary conditions, using equation (1.66) compared with those calculated by Kardas. It is seen that the truncation error is much higher other than at the first point. The accuracy of the re-calculated results were confirmed by a calculation for the problem:

$N = 5 \quad Bi = 10 \quad M = 0.1 \quad \tau = 0.2 \quad \therefore \quad \Delta\tau = 0.004$

The truncation error according to Kardas is: - 0.0000096 and using equation (1.66), 0.00082 (see table B.30). Calculating $\theta(0,0.204)$ from equation 1.77 leads to:

$$\theta(0,0.204) = 0.120427$$

The analytical solution is:

$$\theta(0,0.204) = 0.121243$$

Thus the truncation error is 0.00082 which agrees with that calculated from equation (1.66)

Whilst the method used by Kardas is not directly applicable to most problems since analytical solutions do

not exist, the approach is reasonable. It is feasible to estimate the value of the third derivative at the boundary $X = 0$ by a numerical approximation and then estimate a suitable numerical value for ΔX or $\Delta \tau$ for a specified error from equation (1.66). This was tried and whilst success was achieved for small values of Bi, as Bi was increased it became increasingly difficult to obtain good approximations to the third derivative. This line would be suitable for further research.

Q. The integral method

The integral method, as used by Gay, was used to evaluate the temperature-time history of a plate with radiation and adiabatic boundary conditions for R = 10 and the results were compared with the explicit "exact" solution. Results for both the quadratic and cubic profiles are shown in table B.31.

It can be seen that both the profiles give sufficiently accurate results for the boundary X = 0; the quadratic profile is more accurate than the cubic which is a surprising result. Goodman has also observed this behaviour. The results presented by Gay (19) show the integral method to be less accurate than the other methods at the adiabatic face of the plate. The difference in accuracy between the quadratic and cubic profiles is magnified at this face.

R. Analogue computer solution

The analogue computer was used to evaluate the temperature-time history of a plate for the three boundary conditions described by equations (1.24) - (1.29) and uniform initial temperature distribution. The results are shown in figs. C.37 and C.38 for the step change and radiation cases. The convection boundary case is not shown since sufficient information is conveyed by figs. C.37 and C.38. The analogue computer is seen to be more accurate in the case of radiation than the step change; this is due to a less harsh boundary condition in the former case.

The analogue computer solution is quick and can be adjusted simply for various initial conditions; this is not the case with numerical methods.

Time Sharing

The analogue computer is a useful tool for solving partial differential equations, however, if the thermal conductivity and specific heat are temperature dependent, a large amount of non-linear equipment is required.

Examining equation 1.2 for a plate with step change and adiabatic boundary conditions:

$$\rho C \frac{\partial T}{\partial t} = \frac{\partial}{\partial x}\left(k \frac{\partial T}{\partial x}\right) \qquad (1.2)$$

a difference analogue of equation 1.2 is:

$$\rho C_m \frac{\partial T_m}{\partial t} = \frac{k_{m+\frac{1}{2}}(T_{m+1} - T_m) - k_{m-\frac{1}{2}}(T_m - T_{m-1})}{\Delta x^2} \qquad (2.20)$$

when T and Δx are in dimensionless form, equation (2.20) becomes:

$$\Delta X^2 \frac{\partial \theta_m}{\partial t} = \frac{1}{C_m L^2}\left[\frac{k_{m+\frac{1}{2}}}{\rho}(\theta_{m+1} - \theta_m) - \frac{k_{m-\frac{1}{2}}}{\rho}(\theta_m - \theta_{m-1})\right] \qquad (2.21)$$

If the variation of the temperature of a steel plate is considered, between 500°C and 0°C, the specific heats and thermal conductivities as functions of θ are shown in table 2.2.

The values of C and k given in table 2.2 must be scaled. C is divided by its maximum value (0.157) and K was scaled so that max $(K/\rho C) \neq 1.0$. The scaled values are shown in table 2.3. The scaling introduces a factor of 3.4154 x 0.157. Equation 2.21 now becomes:

$$\Delta X^2 L^2 . 3.4154 \times 0.157 \frac{\partial \theta_m}{\partial t} = \frac{1}{C'_m}\left[\left(\frac{k'}{\rho}\right)_{m+\frac{1}{2}}(\theta_{m+1} - \theta_m) - \left(\frac{k'}{\rho}\right)_{m-\frac{1}{2}}(\theta_m - \theta_{m-1})\right] \qquad (2.22)$$

if $\Delta X = 0.2$ (i.e. 5 increments) and L = 1:

$$\frac{1}{13.405}\frac{\partial \theta_m}{\partial t} = \frac{1}{C'_m}\left[\left(\frac{k'}{\rho}\right)_{m+\frac{1}{2}}(\theta_{m+1} - \theta_m) - \left(\frac{k'}{\rho}\right)_{m-\frac{1}{2}}(\theta_m - \theta_{m-1})\right] \qquad (2.23)$$

where t is in hours.

In the usual analogue computer method, 3 function generators and three multipliers would be required for each section. Therefore, if $\Delta X = 0.2$, as above, 15 function generators and 15 multipliers would be required. The Hybrid-48

TABLE 2.2

THERMAL CONDUCTIVITY AND SPECIFIC HEAT AS FUNCTIONS OF θ

θ	C cal/gm °C	k $\frac{\text{cal . cm}}{\text{cm}^2\ ^\circ\text{C sec}}$
0	0.111	0.130
0.1	0.114	0.122
0.2	0.118	0.114
0.3	0.122	0.108
0.4	0.126	0.104
0.5	0.131	0.101
0.6	0.135	0.097
0.7	0.141	0.094
0.8	0.146	0.092
0.9	0.152	0.088
1.0	0.157	0.086

$\rho = 0.2846\ \text{lb/in}^3$

TABLE 2.3

SCALED VALUES OF C (c') and k/ρ (k'/ρ)

θ	c'	$\frac{k'}{\rho}$
0	0.707	0.707
0.1	0.726	0.662
0.2	0.752	0.618
0.3	0.777	0.587
0.4	0.802	0.563
0.5	0.834	0.550
0.6	0.860	0.529
0.7	0.898	0.512
0.8	0.930	0.502
0.9	0.968	0.478
1.0	1.000	0.467

has 3 function generators and 4 multipliers. The use of a suitable logic program makes it possible to share 2 function generators and 2 multipliers amongst all the sections; the function generators and multipliers are switched into each section in turn by the high speed D/A switches thus the derivatives, $\partial\theta_m/\partial t$ are calculated for each section and stored on track/store amplifiers. The calculation of the derivatives is done while the computer is in HOLD. When all the derivatives have been calculated, the computer is switched to OPERATE for a short time then back to HOLD. The whole cycle is repeated until the desired final value of t is reached.

The equations to be solved can be written:

$$\frac{1}{13.405}\frac{\partial\theta_1}{\partial t} = \frac{1}{C'_1}\left[\left(\frac{k'}{\rho}\right)_{1\frac{1}{2}}(\theta_2-\theta_1) - \left(\frac{k'}{\rho}\right)_{\frac{1}{2}}(\theta_1-\theta_0)\right] \qquad (2.24a)$$

$$\frac{1}{13.405}\frac{\partial\theta_2}{\partial t} = \frac{1}{C'_2}\left[\left(\frac{k'}{\rho}\right)_{2\frac{1}{2}}(\theta_3-\theta_2) - \left(\frac{k'}{\rho}\right)_{1\frac{1}{2}}(\theta_2-\theta_1)\right] \qquad (2.24b)$$

$$\frac{1}{13.405}\frac{\partial\theta_3}{\partial t} = \frac{1}{C'_3}\left[\left(\frac{k'}{\rho}\right)_{3\frac{1}{2}}(\theta_4-\theta_3) - \left(\frac{k'}{\rho}\right)_{2\frac{1}{2}}(\theta_3-\theta_2)\right] \qquad (2.24c)$$

$$\frac{1}{13.405}\frac{\partial\theta_4}{\partial t} = \frac{1}{C'_4}\left[\left(\frac{k'}{\rho}\right)_{4\frac{1}{2}}(\theta_5-\theta_4) - \left(\frac{k'}{\rho}\right)_{3\frac{1}{2}}(\theta_4-\theta_3)\right] \qquad (2.24d)$$

$$\frac{1}{13.405}\frac{\partial\theta_5}{\partial t} = \frac{1}{C'_5}\left[\left(\frac{k'}{\rho}\right)_{4\frac{1}{2}}(\theta_4-\theta_5) - \left(\frac{k'}{\rho}\right)_{4\frac{1}{2}}(\theta_5-\theta_4)\right] \qquad (2.24e)$$

If the integrators for $\partial\theta_m/\partial t$ with a time-scale of 1 sec are used, 1 hr = 13.405 secs.

The analogue and logic circuits for the solution of equations (2.24) are shown as figs. 2.9 and 2.10. The

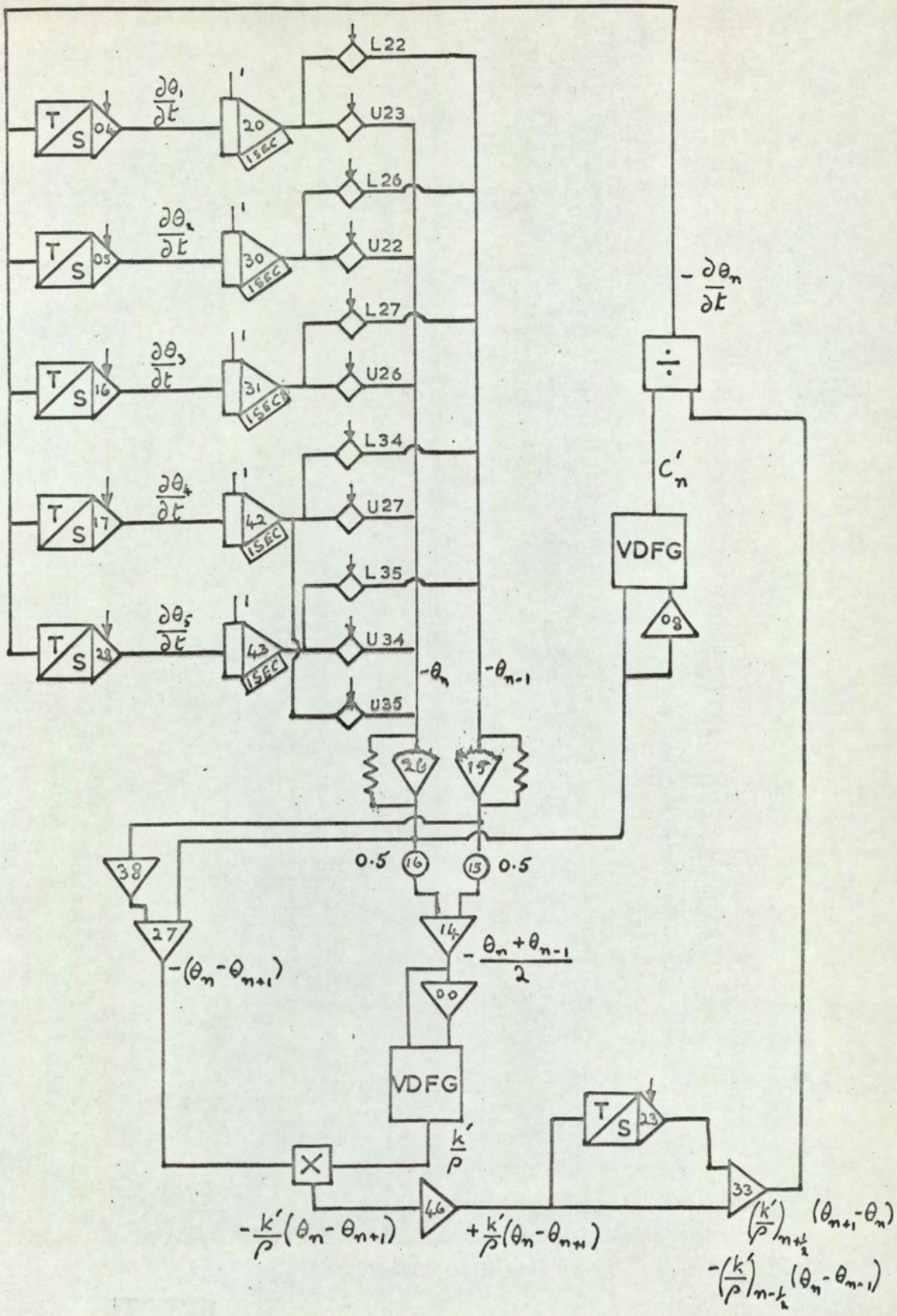


FIG 2.9 TIME SHARING CIRCUIT

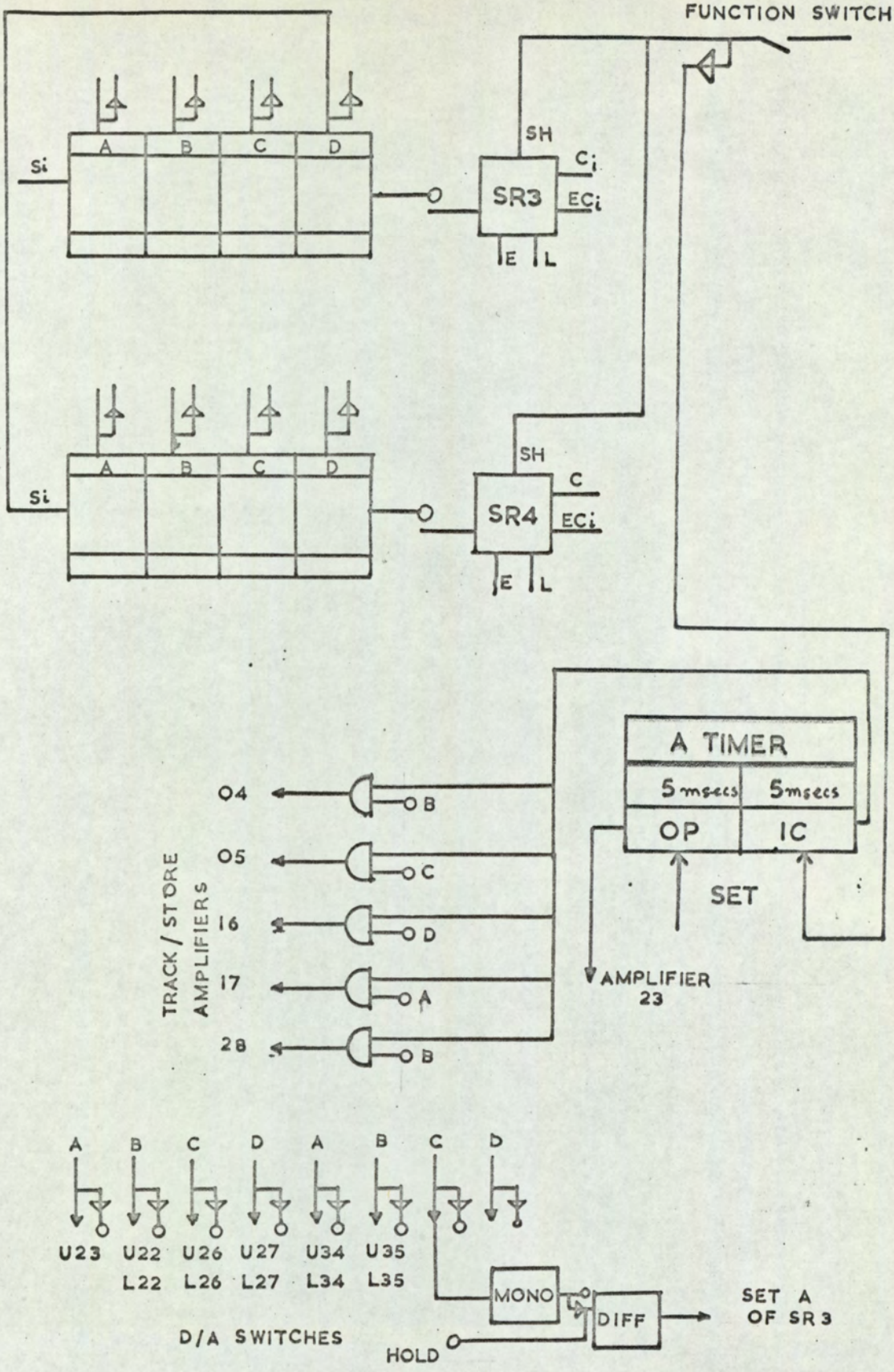


FIG 2.10 LOGIC CIRCUIT

operation is as follows:

When $t < 0$, the inverse output from function switch 2 holds the A timer in IC. The normal output ('0') ensures that SR3 and SR4 (cascaded shift registers) cannot shift. Track/store amplifier 23 is in the store mode. Flip-flop A of SR3 is set manually and the computer is set into OPERATE.

When $t = 0$, function switch 2 is closed. This causes a '1' on SH (shift) of SR3 and SR4 and '0' on SET IC of the A timer. Since the clock is set at 100 cps, and the timer set at 5 msec IC and 5 msec OP, the timer will be in IC for half the clock pulse and OP for the other half. Considering the 5 msec in IC while flip-flop A is set:

D/A switch U23 is conducting which causes θ_1 to appear on the output of amplifier 26 (the amplifier for θ_{n+1}). Since there is no switch conducting on the line for θ_n, $\theta_0 = 0$. The average of θ_1 and θ_0 is calculated hence $(k'/\rho)_{\frac{1}{2}}$ appears on the output of the VDFG; this is multiplied by $(\theta_1 - \theta_0)$ and the inverted result appears as input to amplifier 23 which is currently in the 'store' mode. After 5 msecs the timer switches to OP and amplifier 23 tracks the incoming voltage for 5 msecs.

At the next clock pulse, the timer switches to IC, flip flop A of SR3 is reset and B is set. Switches U22 and L22 are therefore made to conduct which cause θ_2 and θ_1 to

appear as θ_{m+1} and θ_m respectively and $(k'/\rho)_{1\frac{1}{2}}$ appears as the output of the one function generator and C_1' as the output of the other. The expression $-(k'/\rho)_{1\frac{1}{2}}(\theta_2 - \theta_1)$ is the input to amplifier 23 but this is in the 'store' mode, therefore the output is unaffected and remains $(k'/\rho)_{\frac{1}{2}}(\theta_1-\theta_0)$. These two terms are therefore added (and inverted) by amplifier 33 giving:

$$\left(\frac{k'}{\rho}\right)_{1\frac{1}{2}}(\theta_2-\theta_1) - \left(\frac{k'}{\rho}\right)_{\frac{1}{2}}(\theta_1-\theta_0) \qquad (2.25)$$

this is divided by C_1' to produce $\partial\theta_1/\partial t$ (note the constant 1/13.405 is absorbed in the time base) which is tracked by amplifier 04. After 5 msec in IC, the timer switches to OP causing amplifier 04 to store $\partial\theta_1/\partial t$. Amplifier 23 will now track $-(k'/\rho)_{1\frac{1}{2}}(\theta_2-\theta_1)$ After a further 5 msecs, the timer will switch to IC simultaneously with the next clock pulse. Amplifier 23 will store its value and $\partial\theta_2/\partial t$ will be calculated. The whole procedure is repeated until $\partial\theta_5/\partial t$ has been evaluated.

The next clock pulse sets flip-flop C of SR4 resulting on the input of '1' to monostable O. The inverse output will thus be '0' for a preset time interval; this will cause HOLD to be released for this period. When a '1' appears as the inverse output of the monostable, flip-flop A of SR3 is set via a differentiator, thus the whole cycle is repeated until the desired final value of t is reached. The solution

to the problem is shown in fig. C.39.

Accuracy

Since the problem is non-linear, no analytical solution exists hence no direct comparison can be made. It is to be expected that the time sharing method is as accurate as the usual analogue computer method. However, it is to be expected that factors such as integration period and clock frequency will affect the accuracy.

A test on the accuracy of the method was made by over-riding the outputs of the VDFG's to give values of 1 and setting the time base to 25 secs. The problem then reduced to equation (1.168) with step change and adiabatic boundary conditions; thus the results were compared with the analytical solution to this problem given by equation (1.169).

The accuracy of the method for X = 0.2 and 1.0 and various monostable settings is shown in fig. C.40.

Monostable time and clock frequency

In the normal analogue computer method, the value of $\partial\theta_n/\partial t$ is constantly changing since it is calculated from the instantaneous values of θ_{n-1}, θ_n and θ_{n+1}. In the time-sharing method, it is calculated from values of θ_{n-1} θ_n and θ_{n+1} at time t and this value is integrated for a

short period of time, Δt i.e. $\partial\theta_n/\partial t$ is held constant over Δt. It is to be expected that the smaller Δt is made, the more accurate will be the solution. The disadvantage of making Δt small is that a large number of integration steps are required, hence if the clock frequency is low it will take a long time to obtain a solution. If the clock frequency is high, Δt can be made small but a high clock frequency produces inaccurate results. It is necessary to use an integrate period which gives accurate results without making the computation too lengthy. The accuracy for various monostable settings is demonstrated in fig. C.40. The solution time as a function of monostable setting is shown in table B.32. The highest permissible clock frequency was found to be 100 c.p.s; higher frequencies made the solution inaccurate.

SECTION 3

CONCLUSIONS

Conclusions

When a one-dimensional heat conduction problem is considered it is obvious that the Dusinberre explicit method has advantages as far as computation time and accuracy are concerned but since the trend has always been to make faster computers, the superior stability of the implicit method cannot be ignored. In this work, it has been found that the backward difference method is most accurate.

When one is considering two-dimensional problems subject to radiation boundary conditions, it appears that the alternating direction implicit method is efficient. The fully implicit method should not be ignored since the method of partitioning the matrix of coefficients reduces the iteration time. Although it is known that the alternating direction implicit method is stable, one must avoid using too large a time step since if the temperature of the point under consideration is varying rapidly, an explicit formulation (used to formulate the right hand side in the A.D.I.P.) will result in a large error being introduced with the consequence that meaningless results are obtained. The fully implicit method does not have this disadvantage (see fig. C.31).

The solution of the three-dimensional problems with radiation boundary conditions is best accomplished by the

alternating direction implicit method described in part N of section 2. Care must again be taken to avoid a meaningless explicit formulation of the right-hand side of the equations. The method is useful when the temperatures are varying slowly. A fully implicit method cannot be recommended due to the intolerable amount of computer storage required.

Only a small amount of attention has been paid to the integral and analogue computer solutions. The integral method using a quadratic profile gives accurate results for the case considered. The cubic profile is not so successful. The analogue computer solution is useful and accurate but has the disadvantage of requiring large amounts of equipment. The use of the parallel hybrid computer removes this disadvantage to an extent thus making the method practicable.

This work is by no means exhaustive but does give a useful guide for the solution of heat conduction problems. It would now be worthwhile to make further experiments, similar to those done with the one-dimensional case, with the three-dimensional alternating direction implicit method. Further, the investigation of this method when the specific heat and thermal conductivity are considered as temperature dependent would be a useful contribution.

Considerable emphasis is being put on computer-aided design nowadays, therefore a method of presenting all kinds

of heat conduction problems to a computer in a form both suitable to man and machine could be a significant advance.

SECTION 4

APPENDICES

APPENDIX A

COMPUTER PROGRAMS

EXPLICIT METHOD FOR A PLATE WITH RADIATION AND ADIABATIC BOUNDARY CONDITIONS

```
   +0,10&
0,10301,4:6410,4
   6410,4:203,4
   262,4:3012,3
   204,4:2611,4
   302,3:000
5, 0011,4/200,5
   3011,4:0513,4
   468,10:2211,4
   404,10:300,5
   630,5:630,5
10,630,5:630,4
   6410,4:636,3
   610,5:610,5
   601,5:601,5
   631,4:600,5
15,2021,5:2611,4
   2211,4:000
   0011,4/30-1,5
   0011,4/610,5
   0011,4/610,5
20,0011,4/601,5
   631,4:000
   0011,4/600,5
   0011,4/2021,5
   3011,4:0518,4
25,4626,10:2211,4
   4017,10:000
   0018,4/300,5
   0018,4/600,5
29,400,11:000
   *
0,11 0013,4/610,5
   0013,4/610,5
   631,4:000
   0013,4/600,5
   0013,4/2021,5
5, 302,4:603,4
   202,4:614,4
   4522,11:7429
   7430:7431
   7420:741
10,7421:7427
   746:302,4
   730,2:403,2
   000:003
   7428:7428
15,7428:3021,5
   730,2:403,2
   000:008
```

$\Delta\tau = M/N^2$

$\tau = 0$

$L = 0.0395$

$\theta_{(n,0)} = 1 \qquad n = 0, (1), N$

$-2R\Delta x\,\theta^4_{(0,\tau)}$

$\theta_{(0,\tau+\Delta\tau)} = \theta_{(0,\tau)} + M(2\theta_{(1,\tau)} - 2R\Delta x\,\theta^4_{(0,\tau)})$

$n = 1$

$\theta_{(n,\tau+\Delta\tau)} = \theta_{(n,\tau)} + M(\theta_{(n-1,\tau)} - 2\theta_{(n,\tau)} + \theta_{(n+1,\tau)})$

$n = 1, (1), N-1$

$\theta_{(N,\tau+\Delta\tau)} = \theta_{(N,\tau)} + M(2\theta_{(N-1,\tau)} - 2\theta_{(N,\tau)})$

$\tau = \tau + \Delta\tau$

$(\tau - L)$

if $\tau > L$, print τ, $\theta_{(0,\tau+\Delta\tau)}$, $\theta_{(N,\tau+\Delta\tau)}$

```
   0013,4/3021,5
   730,2:403,2
20,000:008
   304,4:6013,3
   204,4:2611,4
   0011,4/3021,5
   0011,4/200,5
25,301184:0513,4
   4627,11:2211,4
   4023,11:302,4
   617,3:458,10
   400,12:000
   *
   008,3/7429
   008,3/7430
   008,3/7431
   008,3/745
   008,3/7414
   008,3/744
   008,3/7428
   008,3/7415
   008,3/746
   008,3/7428
   008,3/7416
   008,3/7418
   008,3/7415
   008,3/747
   008,3/7418
   008,3/741
   008,3/7413
   7429:7430
   7427:7413
   741:7428
   7428:7429
   7430:7430
   000:000
   000:000
   700:4526,12
   700:4527,12
   4025,12:700
   4526,12:400,13
   *
   730,1:402,1
   730,2:403,2
   000:006
   7429:7430
   7430:7430
   7430:7430
   400,8:000
   400,8:000(
```

$\theta_{(n,\tau)} = \theta_{(n,\tau+\Delta\tau)}$ $n = 0, (1), N$

if $\tau < 0.95$, calculate $\theta_{(n,\tau+\Delta\tau)}$ for next time s

prints "END OF PROGRAM"

reads computation time and outputs it at end of numerical results

BACKWARD DIFFERENCE METHOD USING SIMPLE ITERATION

```
0,10 301,4:6410,4
     6410,4:203,4
     262,4:3012,3
     204,4:2611,4
     302,3:000
  5, 0011,4/200,5
     3011,4:0513,4
     468,10:2211,4
     404,10:300,5
     2017,4:301,4
 10, 601,4:602,3
     208,4:2611,4
     0011,4/300,5
     0011,4/2063,5
     0011,4/2084,5
 16, 3011,4:0513,4
     4617,10:2211,4
 17, 4012,10:400,11
     *

0,11 3017,4:6317,4
     6317,4:631,4
     630,4:634,3
     6410,4:601,4
     601,4:602,3
  5, 2042,5:2611,4
     2211,4:3011,4
     050,3:468,11
     4410,11:301,4
     636,3:205,4
 10, 4012,11:301,4
     633,3:205,4
     3011,4:0513,4
     4214,11:4016,11
     301,4:636,3
 15, 206,4:4417,11
     301,4:633,3
     206,4:306,4
     0011,4/6441,5
     0011,4/20105,5
 20, 400,12:000
     *

0,12 635,4:633,3
     608,4:000
     0011,4/2042,5
     0011,4/30105,5
     0011,4/6362,5
  5, 633,3:000
     0011,4/6084,5
  7, 0011,4/2063,5
```

$\Delta\tau = M/N^2$

$\tau = 0$

$L = 0.0395$

$\theta_{(n,0)} = 1.0 \qquad n = 0, (1), N$

$\theta^{(m+1)}_{(0,\tau+\Delta\tau)} = \theta_{(0,\tau)}$

calculates $(1+2M)$

$d_n = \theta_{(n,\tau)}$ see equation (1.94)

$\delta_n = d_n \quad (\delta_0 = d_0)$

2. M. R. $\theta^{(m+1)}_{(0,\tau+\Delta\tau)}$ 3

$\beta_0 = (1+2M) + 2.M.R.\Delta\tau\,\theta^{(m+1)}_{(0,\tau+\Delta\tau)}$ 3

$n = 1$

if $n = 1$, $C = -2.M$ else $C = -M$

if $n = N$, $A = -2.M$ else $A = -M$

A/β_{n-1}

$\beta_n = b - C.A/\beta_{n-1}$

$\delta_n = d_n - A.\delta_{n-1}/\beta_{n-1}$

```
    3011,4:0513,4
    4211,12:2211,4
10, 446,11:000          if n ≠ N , n=n+1 , repeat loop from 6,11
    0013,4/3063,5
    0013,4/6442,5
    0013,4/2021,5       θ(N,τ+δτ)^(m+1) = δ_N / β_N     see equation (1.100)
    3013,4:2011,4       n = N
15, 3011,4:050,3
    4217,12:4019,12     if n = 1 , c = 2  else c = 1
    301,4:636,3
    205,4:4420,12
    301,4:633,3
20, 205,4:400,13        θ(n-1,τ+δτ)^(m+1) = [δ_{n-1} - c θ(n,τ+δτ)^(m+1)] / β_{n-1}
    *                       see equation (1.99)
0,13 0011,4/3021,5
    635,4:633,3
    0011,4/6062,5
    0011,4/6441,5
    0011,4/2020,5
5,  3011,4:050,3
    468,13:3011,4
    050,3:2011,4        n = n-1
    4015,12:3017,4
    6121,5:4510,13
10, 4011,13:633,3       | θ(0,τ+δτ)^(m) - θ(0,τ+δτ)^(m+1) | - convergence limit
    619,4:4513,13       if not within convergence limit, θ(0,τ+δτ)^(m) = θ(0,τ+δτ)^(m+1) and a
    3021,5:2017,4       further iteration is performed.
    400,11:302,4
    603,4:202,4         τ = τ + δτ
15, 614,4:459,14
    7429:7430
    7431:7420
    741:7421
    7427:746            if τ > L ,  τ , θ(0,τ+δτ)^(m+1) , θ(N,τ+δτ)^(m+1)  are output
20, 302,4:000
    730,2:403,2
    000:003
23, 400,14:000
    *

    7428:7428
    7428:7428
    3021,5:000
    730,2:403,2
    000:008
    0013,4/3021,5
    730,2:403,2
    000:008
    304,4:6013,3
    204,4:2611,4        L = L + 0.04
    0011,4/3021,5
```

BACKWARD DIFFERENCE METHOD USING NEWTON-RAPHSON ITERATI

```
      +0,10&
0,10  301,4:6410,4
      6410,4:203,4
      262,4:3012,3
      204,4:2611,4
      302,3:000
 5,   0011,4/200,5
      0011,4/2021,5
      3011,4:0513,4
      469,10:2211,4
      404,10:301,4
 10,  601,4:602,3
      2019,4:208,4
 12,  400,11:000
      *

      +0,11&
0,11  3021,5:6321,5
      6321,5:631,4
      630,4:6311,3
      6410,4:6019,4
      2042,5:3021,5
 5,   6321,5:6321,5
      6321,5:630,4
      631,4:634,3
      6410,4:200,6
      3019,4:6321,5
 10,  600,6:200,6
      301,4:6322,5
      636,3:600,6
      610,5:633,3
      2063,5:2084,5
 15,  400,12:000
      *

      +0,12&
0,12  2611,4:2211,4
      3019,4:000
      0011,4/6321,5
      201,6:301,4
      0011,4/6320,5
 5,   633,3:601,6
      202,6:301,4
      0011,4/6322,5
      633,3:602,6
      0011,4/610,5
 10,  633,3:000
      0011,4/2084,5
 12,  400,13:000
      *

      +0,13&
0,13  3011,4:0518,4
```

$\Delta\tau = M/N^2$

$\tau = 0$

$L = 0.0395$

$\theta_{(n,\tau)} = 1 \qquad \theta^{(m)}_{(n,\tau+\Delta\tau)} = 1 \qquad n = 0, (1), N$

calculates $(1+2m)$

$\beta_0 = (1+2M) + 8.M.R.\Delta x.\theta^{(m)\,3}_{(0,\tau+\Delta\tau)}$ see equation (1.104)

$2M.R.\Delta x.\theta^{(m)\,4}_{(0,\tau+\Delta\tau)}$

$(1+2M)\,\theta^{(m)\,*}_{(0,\tau+\Delta\tau)} + 2MR\Delta x\,\theta^{(m)\,4}_{(0,\tau+\Delta\tau)}$

$(1+2m)\,\theta^{(m)}_{(0,\tau+\Delta\tau)} + 2M.R.\Delta x.\theta^{(m)\,4}_{(0,\tau+\Delta\tau)} - 2M\,\theta^{(m)}_{(1,\tau+\Delta\tau)}$

$\delta_0 = d_0 = -\left[(1+2m)\theta^{(m)}_{(0,\tau+\Delta\tau)} + 2M.R.\Delta x.\theta^{(m)\,4}_{(0,\tau+\Delta\tau)} - 2M\,\theta^{(m)}_{(1,\tau+\Delta\tau)} - \theta_{(0}\right.$

$d_n =$

$-\left[\left(-M\,\theta^{(m)}_{(n-1,\tau+\Delta\tau)} + (1+2m)\,\theta^{(m)}_{(n,\tau+\Delta\tau)} - M\,\theta^{(m)}_{(n+1,\tau+\Delta\tau)}\right) - \theta_{(n,\tau)}\right]$

```
      462,13:2211,4
      401,12:3019,4
      0013,4/6321,5
      203,6:301,4
 5,   0018,4/6321,5
      636,3:603,6
      0013,4/610,5
      633,3:000
      0013,4/2084,5
10,   400,14:000
      *
      +0,14&
0,14  2611,4:2211,4
      3011,4:050,3
      423,14:405,14
      301,4:636,3
      205,4:446,14
 5,   301,4:633,3
      205,4:3011,4
      0513,4:468,14
      4410,14:301,4
      636,3:206,4
10,   4012,14:301,4
      633,3:206,4
12,   400,15:000
      *
      +0,15&

0,15  306,4:000
      0011,4/6441,5
      0011,4/20105,5
      635,4:633,3
      608,4:000
 5,   0011,4/2042,5
      0011,4/30105,5
      0011,4/6362,5
      633,3:000
      0011,4/6084,5
10,   0011,4/2063,5
      3011,4:0513,4
      4613,15:2211,4
12,   401,14:400,16
      *
      +0,16&
0,16  0013,4/3063,5
      0013,4/6442,5
      0013,4/20126,5
      3013,4:2011,4
      3011,4:050,3
 5,   426,16:408,16
```

$-2M\,\theta^{(m)}_{(N-1,\tau+\Delta\tau)} + (1+2M)\,\theta^{(m)}_{(N,\tau+\Delta\tau)}$

$d_N = -\left[\left(-2M\,\theta^{(m)}_{(N-1,\tau+\Delta\tau)} + (1+2M)\,\theta^{(m)}_{(N,\tau+\Delta\tau)}\right) - \theta_{(N,\tau)}\right]$

$n = 1$

if $n = 1$, $c = -2.M$ else $c = -M$

if $n = N$, $a = -2.M$ else $a = -M$

a / β_{n-1}

$\beta_n = b - c \cdot a / \beta_{n-1}$

$\delta_n = d_n - a/\beta_{n-1} \cdot \delta_{n-1} \qquad n = 1, (1), N$

$\Delta\theta^{(m)}_{(N,\tau+\Delta\tau)} = \delta_N / \beta_N$

$n = N$

```
     4517,19:7429
     7430:7431
     7420:741
     7421:7427
     746:302,4
     730,2:403,2
     000:003
     7420:7420
     7420:7428
     3021,5:000
     730,2:403,2
     000:008
     0013,4/3021,5
     730,2:403,2
     000:000
     304,4:6013,3
     204,4:400,20
     *
     +0,20&
0,20 302,4:617,3
     410,11:7429
     7430:7427
     7413:741
     7429:7430
  5, 7430:000
     000,3/7429
     000,3/7430
     000,3/7431
     008,3/745
 10, 000,3/7414
     000,3/744
     008,3/7428
     000,3/7415
     008,3/746
 15, 000,3/7428
     008,3/7416
     000,3/7410
     008,3/7415
     000,3/747
 20, 008,3/7410
     000,3/741
     000,3/7413
 23, 400,21:000
     *
     +0,21&
     700:452,21
     700:453,21
     401,21:700
     452,21:000
     730,1:402,1
     730,2:403,2
     000:000
     7429:7430
```

if $\tau > L$, print τ, $\theta^{(m+1)}_{(0,\tau+\Delta\tau)}$, $\theta^{(m+1)}_{(N,\tau+\Delta\tau)}$

$L = L + 0.04$

if $\tau > 0.96$, end program

prints "END OF PROGRAM"

inputs computation time and prints it on end of results tape

```
      301,4:636,3
      205,4:449,16
      301,4:633,3
      205,4:000
   10, 0011,4/30126,5
      635,4:633,3
      0011,4/6062,5
      0011,4/6441,5
      0011,4/20125,5
   15,3011,4:050,3
      4618,16:3011,4
      050,3:2011,4
   18,404,16:400,17
      *
0,17 +0,17&
      2611,4:000
      0011,4/30126,5
      0011,4/6021,5
      0011,4/2021,5
    5, 3011,4:0513,4
      466,17:2211,4
      401,17:30126,5
      207,4:2611,4
      2211,4:307,4
   10, 4110,17:4410,17
      633,3:204,6
      0011,4/30126,5
      4113,17:4413,17
      633,3:614,6
   15, 4516,17:000
      0011,4/30126,5
   17, 207,4:400,18
      *
      +0,18&
 0,18 3011,4:0513,4
      462,18:2211,4
      448,17:307,4
      414,18:444,18
      633,3:619,4
    5, 416,18:400,11
      302,4:603,4
      202,4:2611,4
      0011,4/3021,5
      0011,4/200,5
   10, 3011,4:0513,4
      4612,18:2211,4
   12, 408,18:400,19
      *
      +0,19&
 0,19 302,4:614,4
```

if $n = 1$, $c = -2M$
else $c = -M$

$$\Delta\theta^{(m)}_{(n-1,\tau+\Delta\tau)} = (\delta_{n-1} - c\,\Delta\theta^{(m)}_{(n,\tau+\Delta\tau)})/\beta_{n-1}$$

$$\theta^{(m+1)}_{(n,\tau+\Delta\tau)} = \theta^{(m)}_{(n,\tau+\Delta\tau)} + \Delta\theta^{(m)}_{(n,\tau+\Delta\tau)}$$

finds $\max |\Delta\theta^{(m)}_{(n,\tau+\Delta\tau)}|$ - convergence limit

if convergence is not satisfied, perform another iteration

$$\theta_{(n,\tau)} = \theta^{(m+1)}_{(n,\tau+\Delta\tau)} \qquad n = 0, (1), N$$

$\tau - L$

```
=BACKWARD METHOD FOR A PLATE WITH RADIATION AND ADIABATIC BOUNDARY
CONDITIONS USING RELAXATION
0,10 301,4:6410,4
     6410,4:203,4
     262,4:3012,3
     204,4:2611,4
     302,3:000
5,   0011,4/200,5
     3011,4:0513,4
     468,10:2211,4
     404,10:301,4
     601,4:602,3
10,  2019,4:2611,4
     0011,4/300,5
     0011,4/2021,5
     3011,4:0513,4
     4615,10:2211,4
15,  4011,10:3021,5
     6321,5:6321,5
     6410,4:630,4
     631,4:634,3
     6019,4:200,6
20,  3022,5:631,4
     634,3:600,5
     640,6:20231,5
     2611,4:2211,4
     0011,4/3022,5
25,  0011,4/60230,5
     631,4:000
     0011,4/600,5
22,  6419,4:400,11
     *
0,11 0011,4/20231,5
     3011,4:0518,4
     463,11:2211,4
     4024,10:000
     0013,4/30230,5
5,   631,4:634,3
     0013,4/600,5
     6419,4:000
     0013,4/20231,5
     2611,4:000
10,  0011,4/30231,5
     0011,4/6121,5
     6320,4:000
     0011,4/6021,5
     0011,4/20231,5
15,  3011,4:0513,4
     4617,11:2211,4
     4010,11:000
     30231,5:6121,5
```

$\Delta\tau = M/N^2$

$\tau = 0$

$L = 0.0395$ (L controls value of τ for which results prin

$\theta_{(n,0)} = 1.0 \quad n = 0, (1), N$

calculates (1+2M)

$\theta^{(m)}_{(n,\tau+\Delta\tau)} = \theta_{(n,\tau)}$

$\theta^{3}_{(0,\tau+\Delta\tau)}$

$2M.R.\Delta x.\theta^{(m)}_{(0,\tau+\Delta\tau)} + (1+2M)$

$$\bar{\theta}^{(m+1)}_{(0,\tau+\Delta\tau)} = \frac{\theta_{(0,\tau)} + 2M.\theta^{(m)}_{(1,\tau+\Delta\tau)}}{(1+2M) + 2M.R.\Delta x.\theta^{3}_{(0,\tau+\Delta\tau)}}$$

$$\bar{\theta}^{(m+1)}_{(n,\tau+\Delta\tau)} = \frac{(\theta^{(m)}_{(n+1,\tau+\Delta\tau)} + \bar{\theta}^{(m+1)}_{(n-1,\tau+\Delta\tau)})M + \theta_{(n,\tau)}}{(1+2M)}$$

for $n = 1, (1), N-1$

$$\bar{\theta}^{(m+1)}_{(N,\tau+\Delta\tau)} = \frac{2.M.\bar{\theta}^{(m+1)}_{(N-1,\tau+\Delta\tau)} + \theta_{(N,\tau)}}{(1+2M)}$$

$$\theta^{(m+1)}_{(n,\tau+\Delta\tau)} = \theta^{(m)}_{(n,\tau+\Delta\tau)} + \omega\left[\bar{\theta}^{(m+1)}_{(n,\tau+\Delta\tau)} - \theta^{(m)}_{(n,\tau+\Delta\tau)}\right]$$

for $n = 0, (1), N$.

$\theta^{(m+1)}_{(0,\tau+\Delta\tau)} - \theta^{(m)}_{(0,\tau+\Delta\tau)}$

```
        4120,11:4420,11
20,     633,3:207,4
        2611,4:2211,4
        0011,4/30231,5
        0011,4/6121,5
        4125,11:4425,11
25,     633,3:201,6
        617,4:4128,11
        301,6:207,4
        3011,4:0513,4
        4630,11:2211,4
30,     4022,11:000
        307,4:619,4
        4138,11:2611,4
        0011,4/30231,5
        0011,4/2021,5
35,     3011,4:0513,4
        4637,11:2211,4
        4033,11:4415,10
        302,4:603,4
40,     202,4:614,4
        4511,12:7429
        7430:7431
        7420:741
        7421:7427
45,     746:400,12
        *


0,12    302,4:000
        730,2:403,2
        000:003
        7428:7428
        7428:30231,5
5,      730,2:403,2
        000:008
        0013,4/30231,5
        730,2:403,2
        000:008
10,     304,4:6013,3
        204,4:2611,4
        0011,4/30231,5
        0011,4/200,5
        3011,4:0513,4
15,     4616,12:2211,4
        4012,12:302,4
        617,3:4510,10
        7429:7430
        7427:7413
20,     741:7429
        7430:7430
22,     400,13:000
        *
```

$|\theta^{(m+1)}_{(0,\tau+\Delta\tau)} - \theta^{(m)}_{(0,\tau+\Delta\tau)}|$

calculates $\max|\theta^{(m+1)}_{(n,\tau+\Delta\tau)} - \theta^{(m)}_{(n,\tau+\Delta\tau)}|$ for $n = 0, (1), N$

$(\max|\theta^{(m+1)}_{(n,\tau+\Delta\tau)} - \theta^{(m)}_{(n,\tau+\Delta\tau)}|)$ – convergence limit

$\theta^{(m)}_{(n,\tau+\Delta\tau)} = \theta^{(m+1)}_{(n,\tau+\Delta\tau)}$ for $n = 0, (1), N$.

Transfers to 15, 10 to perform another iteration cycle

$\tau = \tau + \Delta\tau$

$\tau - L$

if $\tau > L$ then τ, $\theta^{(m+1)}_{(0,\tau+\Delta\tau)}$ and $\theta^{(m+1)}_{(N,\tau+\Delta\tau)}$ will be output.

$L = L + 0.04$

$\theta_{(n,\tau)} = \theta^{(m+1)}_{(n,\tau+\Delta\tau)}$ for $n = 0, (1), N$

if $\tau < 0.95$ then calculation is performed for next time step

END.

Block 13 merely prints "END OF PROGRAM".

```
0011,4/200,5
3011,4:0513,4
4614,14:2211,4
4010,14:302,4
617,3:4511,10
7429:7430
7427:7413
741:7429
7430:7430
400,15:000
*
```

$\theta_{(n,\tau)} = \theta^{(m+1)}_{(n,\tau+\Delta\tau)}$

if $\tau > 0.95$, program ends
else calculation is performed for a further value of τ

```
008,3/7429
008,3/7430
008,3/7431
008,3/745
008,3/7414
008,3/744
008,3/7428
008,3/7415
008,3/746
008,3/7428
008,3/7416
008,3/7418
008,3/7415
008,3/747
008,3/7418
008,3/741
008,3/7413
400,16:000
*
```

prints "END OF PROGRAM"

```
700:452,16
700:453,16
401,16:700
452,16:000
730,1:402,1
730,2:403,2
000:006
7429:7430
7430:7430
7430:7430
400,8:000
400,8:000(
```

computation time is read from paper tape and printed out.

APPENDIX B

NUMERICAL RESULTS

TABLE B.1

COMPUTATION TIMES TO OBTAIN TEMPERATURE-TIME HISTORY OF A PLATE WITH STEP CHANGE AND ADIABATIC BOUNDARY CONDITIONS. PROGRAMS WRITTEN IN ALGOL.

N	M	Computation Time (Secs) for $\tau \geq .1$	
		Explicit	Crank Nicolson
5	0.1	42.8	65.0
	0.2	33.0	43.5
	0.3	27.8	32.5
	0.4	26.8	31.0
	0.5	25.2	29.0
10	0.1	188.8	417.0
	0.2	106.5	229.0
	0.3	75.2	148.0
	0.4	61.8	130.0
	0.5	55.0	116.0

TABLE B.2

ANALYTICAL SOLUTION FOR A PLATE WITH STEP CHANGE AND ADIABATIC BOUNDARY CONDITIONS.

τ	$\theta(0.2,\tau)$	$\theta(0.4,\tau)$	$\theta(0.6,\tau)$	$\theta(0.8,\tau)$	$\theta(1,\tau)$
0.02	0.6826897	0.9545001	0.9973002	0.9999366	0.999998
0.04	0.5204999	0.8427011	0.9662968	0.9935159	0.9991861
0.08	0.3829181	0.6826263	0.8659207	0.9518003	0.9751617
0.12	0.3166771	0.5846941	0.7750623	0.8832242	0.9175470
0.16	0.2749643	0.5158443	0.6978320	0.8088074	0.8458012
0.20	0.2442480	0.4616467	0.6304015	0.6685362	0.7723137
0.24	0.2192915	0.4159083	0.5703919	0.6685800	0.7022012
0.28	0.1978590	0.3758529	0.5164711	0.6063436	0.6372243
0.32	0.1789234	0.3401248	0.4680303	0.5496050	0.5777551
0.36	0.1619690	0.3080016	0.4237846	0.4980525	0.5236287
0.40	0.1466906	0.2789876	0.3839346	0.4512862	0.4744879
0.44	0.1328822	0.2527400	0.3478427	0.4088906	0.4299237
0.48	0.1203826	0.2289756	0.3151480	0.3704693	0.3895306

TABLE B.3

INACCURACY OF THE EXPLICIT METHOD USING A HIGH LATTICE DENSITY FOR A PLATE WITH STEP CHANGE AND ADIABATIC BOUNDARY CONDITIONS.

N = 50 M = 0.05

τ	$\theta(0.2,\tau)$ %	$E(0.2,\tau)$ %
0.02	0.690579	1.14
0.04	0.549004	5.47
0.06	0.468656	7.40

TABLE B.4

MAXIMUM ERRORS USING THE EXPLICIT METHOD FOR PLATE WITH STEP CHANGE AND ADIABATIC BOUNDARY CONDITIONS

N = 5

M	$\hat{E}(0.2,\tau)$	$\hat{E}(0.4,\tau)$	$\hat{E}(0.6,\tau)$	$\hat{E}(0.8,\tau)$	$\hat{E}(1.0,\tau)$
0.1	0.54	1.1	0.55	0.37	0.41
0.2	0.21	0.42	0.23	0.3	0.2
0.3	0.49	2.5	1	0.76	0.83
0.4	1.9	3.8	2.2	1.4	1.5
0.5	9.9	4.8	3.5	2.2	2.5

TABLE B.5

MAXIMUM ERRORS USING THE CRANK-NICOLSON METHOD FOR A PLATE WITH STEP CHANGE AND ADIABATIC BOUNDARY CONDITIONS.

N = 5.

M	$\hat{E}(0.2,\tau)$	$\hat{E}(0.4,\tau)$	$\hat{E}(0.6,\tau)$	$\hat{E}(0.8,\tau)$	$\hat{E}(1.0,\tau)$
0.1	1.7	1.9	1	0.75	0.8
0.2	2.7	1.1	0.78	0.6	0.68
0.3	4	1.2	0.61	0.48	0.48
0.4	5.3	1.7	1.1	1.1	1.1
0.5	8.8	2.3	1.3	1.3	1.3
0.6	11	2	1.6	1.6	1.6
0.7	12	3.8	1.9	1.9	1.9
0.8	14	3.5	2.1	2.1	2.1
0.9	14	4.7	2.4	2.3	2.3
1	15	5.9	2.7	2.6	2.6

TABLE B.6

CRANK NICOLSON METHOD FOR THE HEAT CONDUCTION EQUATION FOR A PLATE WITH STEP CHANGE AND ADIABATIC BOUNDARY CONDITIONS

N = 20

M = 0.1

TAU	THETA AND ERROR %									
.019	.70	.10	.96	-.15	1.0	-.05	1.0	-.00	1.0	-.00
.039	.53	.12	.85	-.03	.97	-.07	1.0	-.03	1.0	-.01
.059	.44	.10	.76	.01	.92	-.04	.98	-.05	.99	-.04
.079	.39	.08	.69	.03	.87	-.02	.95	-.05	.98	-.05
.099	.35	.07	.63	.03	.82	-.02	.92	-.04	.95	-.05
.119	.32	.05	.59	.02	.78	-.01	.88	-.04	.92	-.05
.139	.30	-.14	.55	.02	.74	-.01	.85	-.03	.88	-.04
.159	.28	.11	.52	.01	.70	-.01	.81	-.03	.85	-.04
.179	.26	.03	.49	.01	.66	-.01	.77	-.02	.81	-.03
.199	.24	.02	.46	.01	.63	-.01	.74	-.02	.77	-.02
.219	.23	.01	.44	.01	.60	-.01	.70	-.01	.74	-.02
.239	.22	.01	.42	.00	.57	-.00	.67	-.01	.70	-.01
.259	.21	.01	.40	.00	.54	-.00	.64	-.01	.67	-.01

TAU	THETA AND ERROR %								
.279	.20	.01	.38	.00	.52	.00	.61	-.00	.64	-.00
.299	.19	.01	.36	.01	.49	.00	.58	.00	.61	-.00
.319	.18	.01	.34	.01	.47	.01	.55	.00	.58	.00
.339	.17	.01	.32	.01	.45	.01	.52	.01	.55	.08
.359	.16	.01	.31	.01	.42	.01	.50	.01	.52	-.27
.379	.15	.01	.20	.01	.40	.01	.48	.01	.50	.01
.399	.15	.02	.28	.02	.38	.01	.46	.01	.48	.01
.419	.14	.02	.27	.02	.37	.02	.43	.02	.45	.02
.439	.13	.02	.25	.02	.35	.02	.41	.02	.43	.02

M = 1.0

TAU	THETA AND ERROR %								
.020	.69	1.1	.96	.19	1.0	-.02	1.0	-.00	1.0	-.00
.040	.52	.71	.85	.33	.97	.04	1.0	.17	1.0	-.01
.060	.44	.51	.75	.31	.92	.12	.93	.02	.99	-.01
.080	.38	.40	.68	.28	.87	.14	.95	.04	.98	.01
.100	.35	.33	.63	.19	.82	.14	.92	.06	.95	.03
.120	.32	.28	.59	.22	.78	.14	.88	.08	.92	.06

TAU					THETA AND ERROR %					
.140	.29	.24	.55	.19	.74	.14	.85	.09	.88	.07
.160	.28	.21	.52	.18	.70	.14	.81	.10	.85	.09
.180	.26	.19	.49	.17	.66	.14	.77	.11	.81	.10
.200	.24	.17	.46	.16	.63	.14	.74	.10	.77	.11
.220	.23	.16	.44	.15	.60	.14	.70	.12	.74	.12
.240	.22	.16	.42	.15	.57	.14	.67	.13	.70	.12
.260	.21	.15	.40	.15	.54	.14	.64	.13	.67	.13
.280	.20	.15	.38	.15	.52	.14	.61	.14	.64	.13
.300	.19	.15	.36	.15	.49	.14	.58	.14	.61	.14
.320	.18	.15	.34	.15	.47	.10	.55	.14	.58	.14
.340	.17	.15	.32	.15	.45.	.15	.52	.14	.55	.14
.360	.16	.15	.31	.15	.42	.15	.50	.15	.52	.15
.380	.15	.15	.29	.15	.40	.15	.47	.15	.50	.15
.400	.15	.16	.28	.15	.38	.15	.45	.15	.48	.15
.420	.14	.16	.27	.16	.37	.16	.43	.31	.45	.16
.440	.13	.16	.25	.16	.35	.16	.41	.16	.43	.16

TABLE B.7

ANALYTICAL SOLUTION FOR A PLATE WITH
CONVECTIVE AND ADIABATIC BOUNDARY CONDITIONS

(a) Bi = 0.1

τ	$\theta(0,\tau)$	$\theta(0.2,\tau)$	$\theta(0.4,\tau)$	$\theta(0.6,\tau)$	$\theta(0.8,\tau)$	$\theta(1.0,\tau)$
0.04	0.977827	0.992125	0.998012	0.999658	0.999961	0.999994
0.08	0.968868	0.984506	0.993453	0.997687	0.999300	0.999683
0.12	0.962080	0.978279	0.988810	0.994822	0.997706	0.998535
0.16	0.956407	0.972905	0.984388	0.991578	0.995391	0.996569
0.20	0.951420	0.968085	0.980171	0.988136	0.992580	0.993999
0.24	0.946884	0.963637	0.976105	0.984576	0.989438	0.991017
0.28	0.942652	0.959441	0.972147	0.980943	0.986079	0.987764
0.32	0.938626	0.955419	0.968266	0.977269	0.982580	0.984333
0.36	0.934744	0.951517	0.964442	0.973570	0.978991	0.980787
0.40	0.930962	0.947700	0.960661	0.969860	0.975348	0.977171
0.44	0.927251	0.943945	0.956912	0.966148	0.971673	0.973511
0.48	0.923593	0.940236	0.953191	0.962439	0.967981	0.969827
0.52	0.919974	0.936562	0.949492	0.958737	0.964284	0.966133
0.56	0.916386	0.932916	0.945814	0.955044	0.960587	0.962435

0.60	0.912823	0.929293	0.942153	0.951361	0.956895	0.958740
0.64	0.909282	0.925691	0.938509	0.947691	0.953211	0.955052
0.68	0.905760	0.922107	0.934880	0.944034	0.949537	0.951374
0.72	0.902254	0.918540	0.931267	0.940389	0.945875	0.947706
0.76	0.898765	0.914988	0.927668	0.936758	0.942225	0.944049
0.80	0.895290	0.911451	0.924084	0.933141	0.938588	0.940406
0.84	0.891830	0.907929	0.920514	0.929537	0.934964	0.936776
0.88	0.888383	0.904420	0.916958	0.925947	0.931354	0.933158
0.92	0.884951	0.900926	0.913415	0.922370	0.927757	0.929555
0.96	0.881532	0.897446	0.909887	0.918808	0.924174	0.925965
1.00	0.878127	0.893979	0.906372	0.915259	0.920604	0.922389

(b) Bi = 1.0

0.04	0.809020	0.930095	0.981957	0.996846	0.999636	0.999946
0.08	0.746554	0.869995	0.943702	0.979709	0.993750	0.997138
0.12	0.703782	0.825191	0.907629	0.956314	0.980255	0.987237
0.16	0.670699	0.789212	0.875369	0.931180	0.961513	0.971001
0.20	0.643391	0.758788	0.846135	0.905652	0.939656	0.950642
0.24	0.619800	0.732034	0.819122	0.880238	0.916122	0.927912
0.28	0.598708	0.707791	0.793767	0.855185	0.891823	0.903975
0.32	0.579360	0.685338	0.769697	0.830627	0.867324	0.879564

0.36	0.561281	0.664216	0.746672	0.806640	0.842971	0.855130
0.40	0.544171	0.644133	0.724532	0.783260	0.818972	0.830950
0.44	0.527835	0.624900	0.703172	0.760506	0.795452	0.807189
0.48	0.512149	0.606394	0.682519	0.738379	0.772478	0.783940
0.52	0.497028	0.588531	0.662520	0.716876	0.750088	0.761258
0.56	0.482416	0.571254	0.643138	0.695986	0.728297	0.739167
0.60	0.468272	0.554521	0.624342	0.675697	0.707107	0.717677
0.64	0.454567	0.538302	0.606106	0.655994	0.686514	0.696786
0.68	0.441278	0.522572	0.588411	0.636862	0.666509	0.676487
0.72	0.428388	0.507310	0.571237	0.618287	0.647079	0.656770
0.76	0.415880	0.492500	0.554567	0.600252	0.628210	0.637622
0.80	0.403741	0.478126	0.538386	0.582742	0.609889	0.619027
0.84	0.391958	0.464174	0.522677	0.565743	0.592100	0.600973
0.88	0.380521	0.450630	0.507428	0.549239	0.574829	0.583443
0.92	0.369418	0.437482	0.492624	0.533216	0.558061	0.566424
0.96	0.358640	0.424718	0.478252	0.517661	0.541781	0.549901
1.0	0.348177	0.412328	0.464300	0.502559	0.525976	0.533860

(c) Bi = 10

0.04	0.255396	0.684676	0.908552	0.982473	0.997823	0.999659
0.08	0.188821	0.532599	0.775967	0.911823	0.970576	0.985709
0.12	0.156769	0.450634	0.682370	0.8336790	0.920026	0.945615

0.16	0.136800	0.396944	0.612914	0.768617	0.860173	0.890076
0.20	0.122482	0.357173	0.557212	0.707018	0.798589	0.829255
0.24	0.111175	0.325058	0.509873	0.650929	0.738762	0.768515
0.28	0.101647	0.297613	0.468152	0.599575	0.682161	0.710293
0.32	0.093290	0.273343	0.430612	0.552408	0.629300	0.655566
0.36	0.085791	0.251464	0.396449	0.509018	0.580252	0.604620
0.40	0.078975	0.231533	0.365172	0.469067	0.534892	0.557427
0.44	0.072740	0.213276	0.336445	0.432266	0.493014	0.513818
0.48	0.067016	0.196502	0.310018	0.398360	0.454384	0.473574
0.52	0.061751	0.181070	0.285686	0.367116	0.418767	0.436460
0.56	0.056904	0.166859	0.263273	0.338325	0.385934	0.402244
0.60	0.052439	0.153769	0.242622	0.311793	0.355672	0.370705
0.64	0.048326	0.141708	0.223593	0.287341	0.327782	0.341636
0.68	0.044536	0.130594	0.206058	0.264807	0.302078	0.314846
0.72	0.041043	0.120352	0.189898	0.244041	0.278389	0.290156
0.76	0.037824	0.110913	0.175006	0.224903	0.256558	0.267402
0.80	0.034858	0.102215	0.161282	0.207266	0.236438	0.246433
0.84	0.032124	0.094199	0.148634	0.191012	0.217897	0.227107
0.88	0.029605	0.086812	0.136978	0.176033	0.200809	0.209297
0.92	0.027284	0.080004	0.126236	0.162228	0.185061	0.192884
0.96	0.025144	0.073730	0.116336	0.149506	0.170549	0.177758
1.0	0.023172	0.067948	0.107213	0.137781	0.157174	0.163818

TABLE B.8 ERRORS OF SOLUTION FOR A PLATE WITH ADIABATIC AND CONVECTION BOUNDARY CONDITIONS

$N = 5$; $Bi = 0.1$; $X = 0$

$(\partial\theta/\partial X)_{(0,\tau)} = (\theta_{(1,\tau)} - \theta_{(-1,\tau)})/2\Delta X$

M	EXPLICIT			CRANK - NICOLSON		
	$\hat{E}$	$\hat{\tau}$	P	$\hat{E}$	$\hat{\tau}$	P
0.1	—	—	—	—	—	—
0.2	0.08	0.04	0	0.15	0.04	0
0.3	0.04	0.048	0	—	—	—
0.4	0.04	0.048	0	0.12	0.048	0
0.5	0.26	0.04	0	—	—	—
0.6		UNSTABLE		0.11	0.048	0
0.8		"		0.10	0.064	0
1.0		"		0.11	0.080	0
1.2		"		0.15	0.048	0
1.4		"		0.22	0.056	0

TABLE B.9 ERRORS OF SOLUTION FOR A PLATE WITH ADIABATIC AND CONVECTION BOUNDARY CONDITIONS

$N = 5$; $Bi = 1.0$; $X = 0$

$(\partial\theta/\partial X)_{(0,\tau)} = (\theta_{(1,\tau)} - \theta_{-1,\tau})/2\Delta X$

M	EXPLICIT			CRANK - NICOLSON		
	$\hat{E}$	$\hat{\tau}$	P	$\hat{E}$	$\hat{\tau}$	P
0.1	0.62	0.04	0	—	—	—
0.2	0.31	0.92	0	1.0	0.04	0
0.3	0.43	0.96	0	—	—	—
0.4	1.5	0.048	0	0.72	0.048	0
0.5	20.0	0.96	96	—	—	—
0.6		UNSTABLE		0.59	0.048	0
0.8		"		0.58	0.064	0
1.0		"		2.0	0.040	0
1.2		"		3.1	0.048	4.8
1.4		"		4.0	0.056	5.6

TABLE B.10 ERRORS OF SOLUTION FOR A PLATE WITH ADIABATIC AND CONVECTION BOUNDARY CONDITIONS

$N = 5$; $Bi = 10.0$; $X = 0$

$(\partial\theta/\partial X)_{(0,\tau)} = (\theta_{(1,\tau)} - \theta_{(-1,\tau)})/2\Delta X$

M	EXPLICIT			CRANK - NICOLSON		
	$\hat{E}$	$\hat{\tau}$	P	$\hat{E}$	$\hat{\tau}$	P
0.1	6.1	0.04	16.7	—	—	—
0.2	8.7	0.04	70.8	4.2	0.04	8.3
0.3	90	0.048	96.0	—	—	—
0.4		UNSTABLE		4.7	0.048	8.0
0.5		"		—	—	—
0.6		"		19	0.048	8.0
0.8		"		47	0.064	20.0
1.0		"		90	0.040	36.0
1.2		"		90	0.048	57.1
1.4		"		90	0.056	77.8

TABLE B.11 ERRORS OF SOLUTION FOR A PLATE WITH ADIABATIC AND CONVECTION BOUNDARY CONDITIONS

N = 10 ; Bi = 10.0 ; X = 0

$(\partial\theta/\partial x)_{(0,\tau)} = (\theta_{(1,\tau)} - \theta_{(-1,\tau)})/2\Delta x$

M	EXPLICIT			CRANK - NICOLSON		
	$\hat{E}$	$\hat{\tau}$	P	$\hat{E}$	$\hat{\tau}$	P
0.1	1.5	0.039	0	—	—	—
0.2	2.1	0.040	4.17	0.8	0.04	0
0.3	2.8	0.039	4.17	—	—	—
0.4	53	0.04	25	0.87	0.04	0
0.5		UNSTABLE		—	—	—
0.6		"		0.96	0.042	0
0.8		"		1.5	0.04	0
1.0		"		2.1	0.04	4.0
1.2		"		5.9	0.048	4.0
1.4		"		22	0.042	8.33

TABLE B.12 ERRORS OF SOLUTION FOR A PLATE WITH ADIABATIC AND CONVECTION BOUNDARY CONDITIONS

N = 15 ; Bi = 10.0; X = 0

$(\partial\theta/\partial X)_{(0,\tau)} = (\theta_{(1,\tau)} - \theta_{(-1,\tau)}) / 2\Delta X$

M	EXPLICIT			CRANK - NICOLSON		
	$\hat{E}$	$\hat{\tau}$	P	$\hat{E}$	$\hat{\tau}$	P
0.1	0.76	0.039	0	—	—	—
0.2	1.1	0.039	0	0.46	0.039	0
0.3	1.2	0.04	0	—	—	—
0.4	1.5	0.039	0	0.48	0.039	0
0.5		UNSTABLE		—	—	—
0.6		"		0.38	0.04	0
0.8		"		0.54	0.039	0
1.0		"		0.46	0.04	0
1.2		"		0.28	0.80	0
1.4		"		0.76	0.44	0

TABLE B.13 ERRORS OF SOLUTION FOR A PLATE WITH ADIABATIC AND CONVECTION BOUNDARY CONDITIONS

$N = 5$; $Bi = 0.1$; $X = 0$

$$(\partial\theta/\partial x)_{(0,\tau)} = (\theta_{(0,\tau)} - \theta_{(-1,\tau)})/\Delta X$$

M	EXPLICIT			CRANK - NICOLSON		
	$\hat{E}$	$\hat{\tau}$	P	$\hat{E}$	$\hat{\tau}$	P
0.1	1.4	0.92	0	—	—	—
0.2	1.3	0.8	0	1.4	0.92	0
0.3	1.4	0.924	0	—	—	—
0.4	1.4	0.93	0	1.4	0.93	0
0.5	1.4	0.96	0	—	—	—
0.6		UNSTABLE		1.4	0.94	0
0.8		"		1.4	0.93	0
1.0		"		1.4	0.92	0
1.2		"		1.4	0.96	0
1.4		"		1.4	0.95	0

TABLE B.14 ERRORS OF SOLUTION FOR A PLATE WITH ADIABATIC AND CONVECTION BOUNDARY CONDITIONS

N = 5 ; Bi = 1.0 ; X = 0

$$(\partial\theta/\partial X)_{(0,\tau)} = (\theta_{(0,\tau)} - \theta_{(-1,\tau)})/\Delta X$$

M	EXPLICIT			CRANK - NICOLSON		
	$\hat{E}$	$\hat{\tau}$	P	$\hat{E}$	$\hat{\tau}$	P
0.1	7.5	0.92	95.8	—	—	—
0.2	7.4	0.92	100	7.6	0.04	100
0.3	7.5	0.96	100	—	—	—
0.4	7.4	0.96	100	7.8	0.96	100
0.5	7.3	0.96	100	—	—	—
0.6		UNSTABLE		7.8	0.96	100
0.8		"		7.8	0.96	100
1.0		"		7.8	0.96	100
1.2		"		7.8	0.96	100
1.4		"		7.8	0.95	100

TABLE B.15 ERRORS OF SOLUTION FOR A PLATE WITH ADIABATIC AND CONVECTION BOUNDARY CONDITIONS

N = 5 ; Bi = 10.0; X = 0

$(\partial\theta/\partial X)_{(0,\tau)} = (\theta_{(0,\tau)} - \theta_{(-1,\tau)})/\Delta X$

M	EXPLICIT			CRANK - NICOLSON		
	$\hat{E}$	$\hat{\tau}$	P	$\hat{E}$	$\hat{\tau}$	P
0.1	15	0.04	95.8	—	—	—
0.2	9.4	0.04	25.0	20	0.04	95.8
0.3	4.5	0.048	12.0	—	—	—
0.4	15	0.048	12.0	14	0.048	96
0.5		UNSTABLE		—	—	—
0.6		"		12	0.048	100
0.8		"		13	0.064	100
1.0		"		—	—	—
1.2		"		—	—	—
1.4		"		—	—	—

TABLE B.16 ERRORS OF SOLUTION FOR A PLATE WITH ADIABATIC AND CONVECTION BOUNDARY CONDITIONS

$N = 5$; $Bi = 0.1$; $X = 0$

$$(\partial\theta/\partial X)_{(0,\tau)} = \tfrac{1}{2}\left[(\theta_{(0,\tau)} - \theta_{(-1,\tau)})/\Delta X + (\theta_{(1,\tau)} - \theta_{(-1,\tau)})/2\Delta X\right]$$

M	EXPLICIT			CRANK - NICOLSON		
	$\hat{E}$	$\hat{\tau}$	P	$\hat{E}$	$\hat{\tau}$	P
0.1	0.72	0.92	0	—	—	—
0.2	0.72	0.92	0	0.72	0.92	0
0.3	0.73	0.96	0	—	—	—
0.4	0.73	0.96	0	0.74	0.96	0
0.5	0.73	0.96	0	—	—	—
0.6		UNSTABLE		0.74	0.96	0
0.8		"		0.74	0.96	0
1.0		"		0.74	0.96	0
1.2		"		0.74	0.96	0
1.4		"		0.74	0.952	0

TABLE B. 17 ERRORS OF SOLUTION FOR A PLATE WITH ADIABATIC AND CONVECTION BOUNDARY CONDITIONS

$N = 5$; $Bi = 1.0$; $X = 0$

$$(\partial\theta/\partial X)_{(0,\tau)} = \tfrac{1}{2}\left[(\theta_{(0,\tau)} - \theta_{(-1,\tau)})/\Delta X + (\theta_{(1,\tau)} - \theta_{(-1,\tau)})\right]$$

M	EXPLICIT			CRANK - NICOLSON		
	$\hat{E}$	$\hat{\tau}$	P	$\hat{E}$	$\hat{\tau}$	P
0.1	4.4	0.04	100	—	—	—
0.2	4.0	0.04	100	4.8	0.04	100
0.3	3.6	0.048	100	—	—	—
0.4	3.5	0.96	100	4.5	0.048	100
0.5	3.8	0.04	100	—	—	—
0.6		UNSTABLE		4.4	0.048	100
0.8		"		4.1	0.064	100
1.0		"		4.0	0.08	100
1.2		"		3.9	0.096	100
1.4		"		4.0	0.112	88.9

TABLE B.18 ERRORS OF SOLUTION FOR A PLATE WITH ADIABATIC AND CONVECTION BOUNDARY CONDITIONS

$N = 5$; $Bi = 10.0$; $X = 0$

$$(\partial\theta/\partial x)_{(0,\tau)} = \tfrac{1}{2}\left[(\theta_{(0,\tau)} - \theta_{(-1,\tau)})/\Delta X + (\theta_{(1,\tau)} - \theta_{(-1,\tau)}/2\Delta X\right.$$

M	EXPLICIT			CRANK - NICOLSON		
	$\hat{E}$	$\hat{\tau}$	P	$\hat{E}$	$\hat{\tau}$	P
0.1	2.7	0.04	4.17	—	—	—
0.2	0.73	0.04	0	6.1	0.04	8.33
0.3	2.2	0.084	4	—	—	—
0.4	—	—	—	3.5	0.048	8.0
0.5	—	—	—	—	—	—
0.6		UNSTABLE		6.7	0.048	4.0
0.8		"		20	0.064	12.0
1.0		"		79	0.04	20.0
1.2		"		90	0.048	28.6
1.4		"		90	0.056	44.4

TABLE B.19

ACCURACY OF THE EXPLICIT METHOD FOR A PLATE WITH CONVECTION AND ADIABATIC BOUNDARY CONDITIONS.

Bi = 10 N = 15 M = 0.1

τ	$\theta_{(0,\tau)}$	$E_{(0,\tau)}$ %	$\theta_{(N,\tau)}$	$E_{(N,\tau)}$ %
0.039	0.26	- 0.76	1.	0
0.079	0.19	- 0.47	0.99	- 0.03
0.119	0.16	- 0.34	0.95	- 0.05
0.159	0.14	- 0.27	0.89	- 0.06
0.199	0.12	- 0.24	0.83	- 0.07
0.239	0.11	- 0.22	0.77	- 0.07
0.279	0.10	- 0.21	0.71	- 0.07
0.319	0.09	- 0.21	0.66	- 0.07
0.359	0.09	- 0.20	0.61	- 0.07
0.399	0.08	- 0.20	0.56	- 0.07
0.439	0.07	- 0.20	0.51	- 0.07
0.479	0.07	- 0.20	0.47	- 0.07
0.519	0.06	- 0.20	0.44	- 0.07
0.559	0.06	- 0.20	0.40	- 0.07
0.599	0.05	- 0.20	0.37	- 0.07
0.639	0.05	- 0.20	0.34	- 0.06
0.679	0.04	- 0.20	0.32	- 0.06
0.719	0.04	- 0.20	0.29	- 0.06
0.759	0.04	- 0.20	0.27	- 0.06
0.799	0.03	- 0.20	0.25	- 0.06
0.839	0.03	- 0.20	0.23	- 0.06
0.879	0.03	- 0.20	0.21	- 0.06
0.919	0.03	- 0.20	0.19	- 0.06

TABLE B.20

EXPLICIT "EXACT" SOLUTION FOR A PLATE WITH RADIATION AND ADIABATIC BOUNDARY CONDITIONS.

R = 10 N = 15 M = 0.1

τ	$\theta_{(0,\tau)}$	$\theta_{(N,\tau)}$	τ	$\theta_{(0,\tau)}$	$\theta_{(N,\tau)}$
0.04	0.603987	0.999767	0.56	0.442446	0.663728
0.08	0.565035	0.991932	0.60	0.436005	0.644138
0.12			0.64	0.429774	0.625742
0.16	0.527033	0.940559	0.68	0.4237742	0.608451
0.20	0.514723	0.907749	0.72	0.417903	0.592180
0.24	0.504293	0.874642	0.76	0.412250	0.576855
0.28	0.494979	0.842524	0.80	0.406774	0.562405
0.32	0.486378	0.811951	0.84	0.401470	0.548766
0.36	0.478274	0.783114	0.88	0.396331	0.535879
0.40	0.470549	0.756036	0.92	0.391352	0.523690
0.44	0.463137	0.730656	0.96	0.386525	0.512150
0.48	0.455997	0.706882	1.00	0.3818461	0.501213
0.52	0.449106	0.684608			

TABLE B.21

ITERATION CYCLES REQUIRED TO ESTABLISH CONVERGENCE FOR A PLATE WITH RADIATION AND ADIABATIC BOUNDARY CONDITIONS

N = 5 R = 10 M = 0.2 Convergence = 0.00001

Time Step	Iteration Cycles	
	Crank-Nicolson	Backward Difference
1	9	27
2	6	15
3	6	12
4	6	10
5	6	9
6	6	8
7	6	8
8	5	7

TABLE B.22 COMPARISON OF THE ACCURACY OF APPROXIMATE METHODS EFFICIENCY FOR RADIATION AND ADIABATIC BOUNDARY CONDITIONS

METHOD	ITERATION TECHNIQUE	MODULUS	RADIATING BOUNDARY TEMPERATURE		INSULATED BOUNDARY TEMPERATURE		COMPUTATION
			AVERAGE ERROR%	MAXIMUM ERROR%	AVERAGE ERROR%	MAXIMUM ERROR%	SECS.
EXPLICIT		0.05	0.33	1.25	0.21	0.32	93
		0.1	0.46	1.78	0.35	0.43	56
		0.15	0.68	2.65	0.61	0.75	43
		0.2	1.06	4.65	1.05	1.30	37
		0.25	1.59	7.92	1.61	2.03	33
		0.3	1.99	7.21	2.3	2.9	32
		0.35	3.69	33.9	3.28	4.21	29
		0.40	6.93	49	5.01	6.54	28
CRANK NICOLSON	SIMPLE	0.1	0.39	0.98	0.20	0.355	180
		0.2	0.46	1.38	0.28	0.44	106
		0.3	0.55	1.82	0.46	0.65	82
		0.4	0.80	3.22	0.76	1.02	70
		0.5	1.29	6.09	1.19	1.59	57
		0.6	1.61	6.71	1.78	2.34	52
		0.7	2.23	8.79	2.46	3.23	51
		0.8	2.73	11.1	3.16	4.21	48
		0.9	3.36	13.9	4.08	5.36	47
		1.0	6.93	74.3	4.90	5.59	46

TABLE B.22 Contd.

BACKWARD DIFFERENCE	SIMPLE	0.1	0.31	0.45	0.09	0.41	188
		0.2	0.18	0.58	0.23	0.48	117
		0.3	0.15	1.19	0.37	0.60	103
		0.4	0.26	1.90	0.53	0.68	89
		0.5	0.64	4.17	0.71	0.94	88
		0.6	0.66	3.35	0.83	1.11	90
		0.7	0.65	3.86	0.86	1.20	103
		0.8	0.63	3.2	0.97	1.33	140
CRANK-NICOLSON	NEWTON-RAPHSON	0.1	0.27	1.01	0.16	0.36	256
		0.2	0.36	1.45	0.26	0.44	167
		0.3	0.49	1.94	0.45	0.66	136
		0.4	0.76	3.29	0.75	1.04	116
		0.50	1.19	6.23	1.18	1.60	100
		0.60	1.56	6.62	1.75	2.32	88
		0.7	2.21	8.60	2.45	3.21	77
		0.8	2.76	11.2	3.19	4.22	72
		0.9	3.36	14.0	4.08	5.34	67

TABLE B.22 Contd.

BACKWARD DIFFERENCE	NEWTON-RAPHSON						
		0.1	0.10	0.14	0.15	0.41	280
		0.2	0.08	0.72	0.27	0.49	186
		0.3	0.18	1.20	0.41	0.60	148
		0.4	0.32	2.00	0.53	0.71	128
		0.5	0.51	3.84	0.66	0.89	112
		0.6	0.59	3.73	0.79	1.06	97
		0.7	0.70	3.70	0.90	1.24	86
		0.8	0.79	3.64	1.05	1.41	78
		0.9	0.87	3.63	1.16	1.59	73
		1.0	1.29	9.30	1.26	1.76	67
		1.1	1.40	9.23	1.38	1.93	62
		1.2	1.52	9.18	1.49	2.10	57
		1.3	1.61	9.13	1.62	2.27	54.0
		1.4	1.74	9.10	1.72	2.44	49.0
		1.5	1.85	9.07	1.84	2.60	47.0
		1.6	1.95	9.05	1.96	2.77	44.0
		1.7	2.06	9.03	2.06	2.94	42.0
		1.8	2.17	9.01	2.14	3.10	42.0
		1.9	2.24	9.00	2.32	3.27	40.0
		2.0	2.36	8.98	2.41	3.43	37.0

TABLE B.23

ITERATION CYCLES REQUIRED TO ESTABLISH CONVERGENCE
FOR A PLATE WITH RADIATION AND ADIABATIC BOUNDARY CONDITIONS

N = 5 R = 0.1 CONVERGENCE = 0.00001

M	ITERATION CYCLES FOR FIRST TIME STEP	
	JACOBIAN	GAUSS-SEIDEL
0.1	4	3
0.2	4	3
0.3	6	4
0.4	6	5
0.5	8	5
0.6	8	6
0.7	10	6
0.8	10	7

TABLE B.24

N = 5 R = 1.0 CONVERGENCE = 0.00001

M	ITERATION CYCLES FOR FIRST TIME STEP	
	JACOBIAN	GAUSS-SEIDEL
0.1	6	5
0.2	8	5
0.3	10	6
0.4	13	6
0.5	15	6
0.6	17	6
0.7	19	6
0.8	21	7

TABLE B.25

N = 5 R = 10 CONVERGE = 0.00001

M	ITERATION CYCLES FOR FIRST TIME STEP	
	JACOBIAN	GAUSS-SEIDEL
0.1	17	15
0.2	29	23
0.3	45	27
0.4	67	30
0.5	101	32
0.6	154	32
0.7	258	32
0.8	548	32
1.0	NO CONVERGENCE	30
1.2	"	29
2.0	"	24

TABLE B.26

ITERATION CYCLES REQUIRED TO ESTABLISH CONVERGENCE FOR A PLATE WITH RADIATION AND ADIABATIC BOUNDARY CONDITIONS USING RELAXATION

N = 5 R = 10 M = 0.1

ω	ITERATIONS
0.1	50
0.2	26
0.3	17
0.4	12
0.5	8
0.6	6
0.7	4
0.8	7
0.9	11
1.0	15

TABLE B.27

COMPUTATION TIMES FOR TWO DIMENSIONAL METHODS

N = 5 M = 0.1

METHOD	COMPUTATION TIME (SECS)
EXPLICIT	240
A.D.I.P.	560
A.D.E.P.	250

TABLE B.28

ACCURACIES OF TWO DIMENSIONAL METHODS

N = 5 M = 0.1 X = 0.2 Y = 0.2

τ	ERROR %		
	EXPLICIT	ADIP	ADEP
0.04	- 1.04	+ 2.95	+ 4.82
0.08	- 0.43	2.42	5.09
0.12	- 0.36	1.66	4.43
0.16	- 0.45	1.06	3.97
0.20	- 0.60	0.61	3.85
0.24	- 0.71	0.37	4.09
0.28	- 0.80	0.31	4.59
0.32	- 0.87	0.30	5.27
0.36	- 0.92	0.38	6.06
0.40	- 0.98	0.50	6.90
0.44	- 1.00	0.63	7.80
0.48	- 1.04	0.79	8.72

TABLE B.29

ACCURACY OF THE ALTERNATING DIRECTION EXPLICIT METHOD

N = 5 M = 0.5 X = 0.2 Y = 0.2

τ	ERROR %
0.04	+ 13.39
0.08	14.70
0.12	13.71
0.16	16.13
0.20	17.09
0.24	19.13
0.28	21.90
0.32	25.19
0.36	28.84
0.40	32.70
0.44	36.90
0.48	37.13

TABLE B.30

RE-EVALUATION OF TRUNCATION ERRORS

$X = 0$

τ	TRUNCATION ERROR. 10^{-4}	
	KARDAS	RE-EVALUATED
0	- 87723.417	- 74456.287
0.02	- 19.732	+ 147.57
0.04	- 4.566	66.631
0.06	- 1.862	40.013
0.08	- 0.969	27.432
0.1	- 0.577	20.326
0.2	- 0.097	8.255
0.3	- 0.033	5.531
0.4	- 0.019	4.322
0.5	- 0.015	3.494
0.6	- 0.012	2.844
0.7	- 0.009	2.318
0.8	- 0.008	1.89
0.9	- 0.006	1.541
1	- 0.005	1.256

TABLE B.31

THE ACCURACY OF THE INTEGRAL METHOD FOR
A PLATE WITH RADIATION AND ADIABATIC BOUNDARY CONDITIONS

τ	EXPLICIT N=15 M = 0.1 $\theta(o,\tau)$	QUADRATIC PROFILE $\theta(o,\tau)$	ERROR %	CUBIC PROFILE $\theta(o,\tau)$	ERROR %
0.001	0.792	0.800	1.01	0.800	1.01
0.007	0.702	0.700	0.29	-	-
0.008	0.695	-	-	0.700	0.72
0.042	0.601	0.600	0.17	-	-
0.047	0.594	-	-	0.600	1.00
0.055	0.586	-	-	0.592	1.02
0.106	0.549	0.548	0.18	-	-
0.294	0.491	0.500	1.83	-	-
0.329	0.484	-	-	0.500	3.30
0.806	0.406	-	-	0.400	1.48
0.845	0.401	0.400	0.25	-	-

TABLE B.32

COMPUTATION TIMES FOR VARIOUS MONOSTABLE SETTINGS

MONOSTABLE m secs.	COMPUTATION TIME (Secs)
10	265
100	58
400	35
500	25

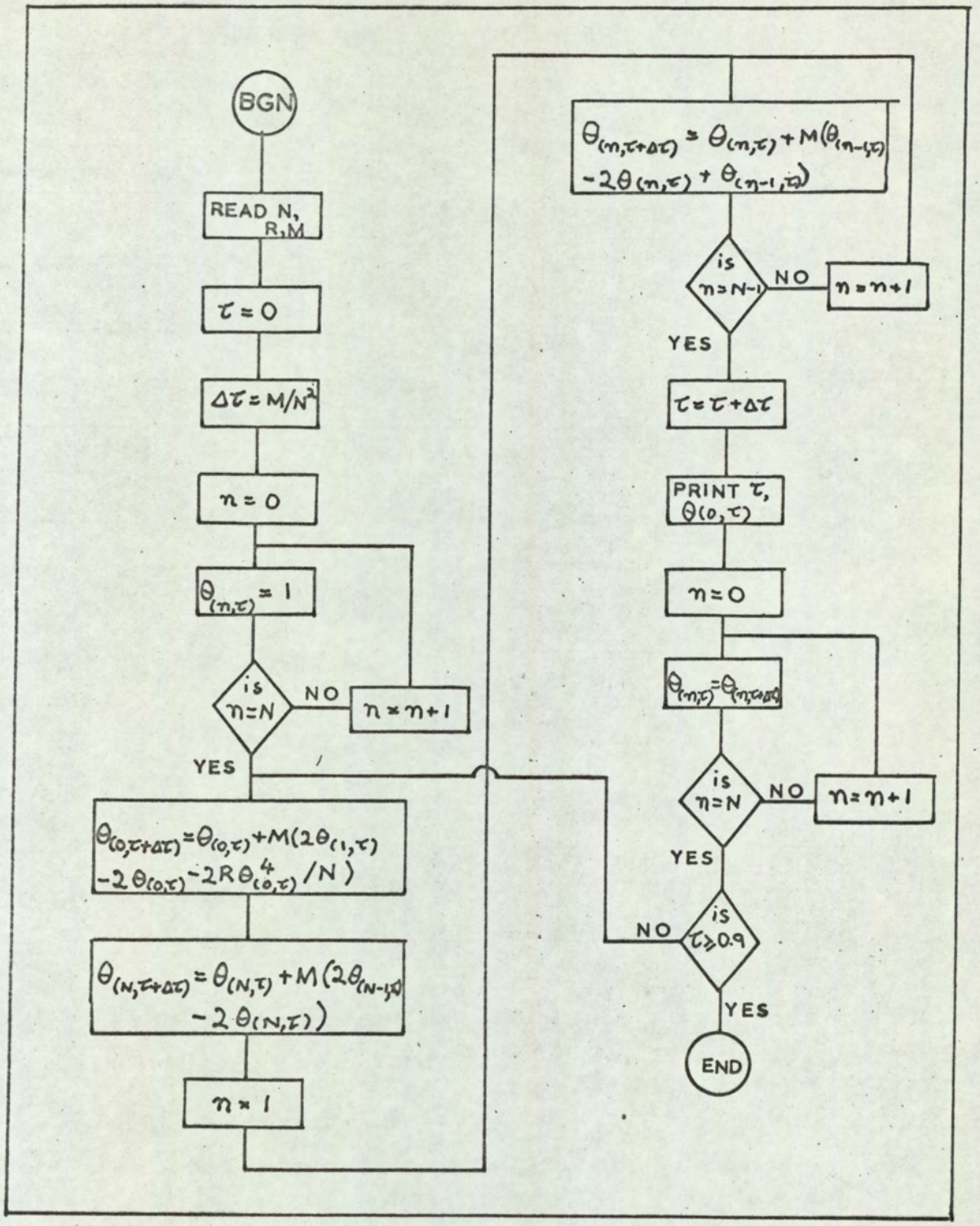


FIG C.1 FLOW SHEET FOR EXPLICIT METHOD WITH RADIATION BOUNDARY CONDITION

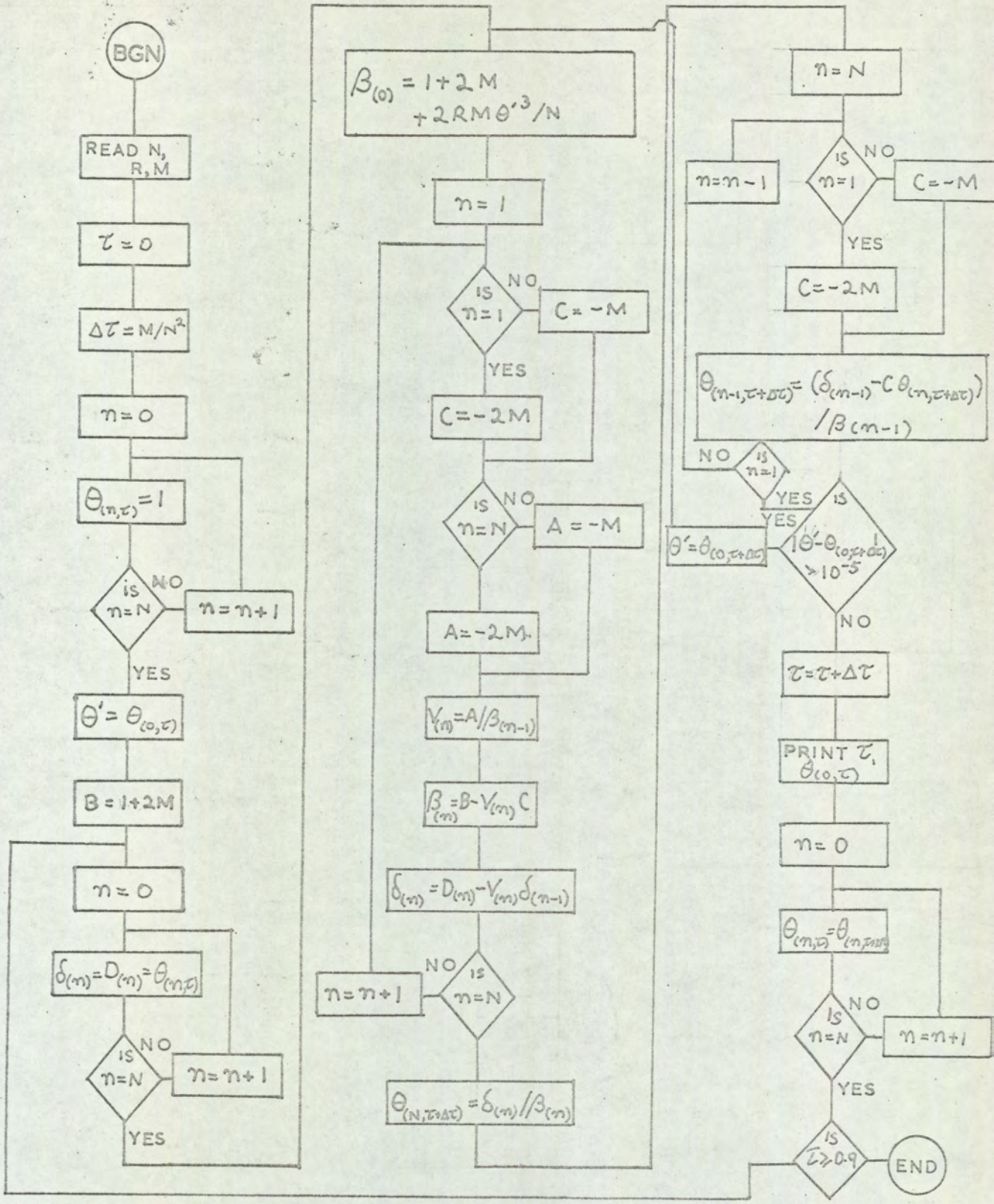


FIG C.2 FLOW SHEET FOR BACKWARD DIFFERENCE METHOD USING SIMPLE ITERATION. RADIATION BOUNDARY CONDITION.

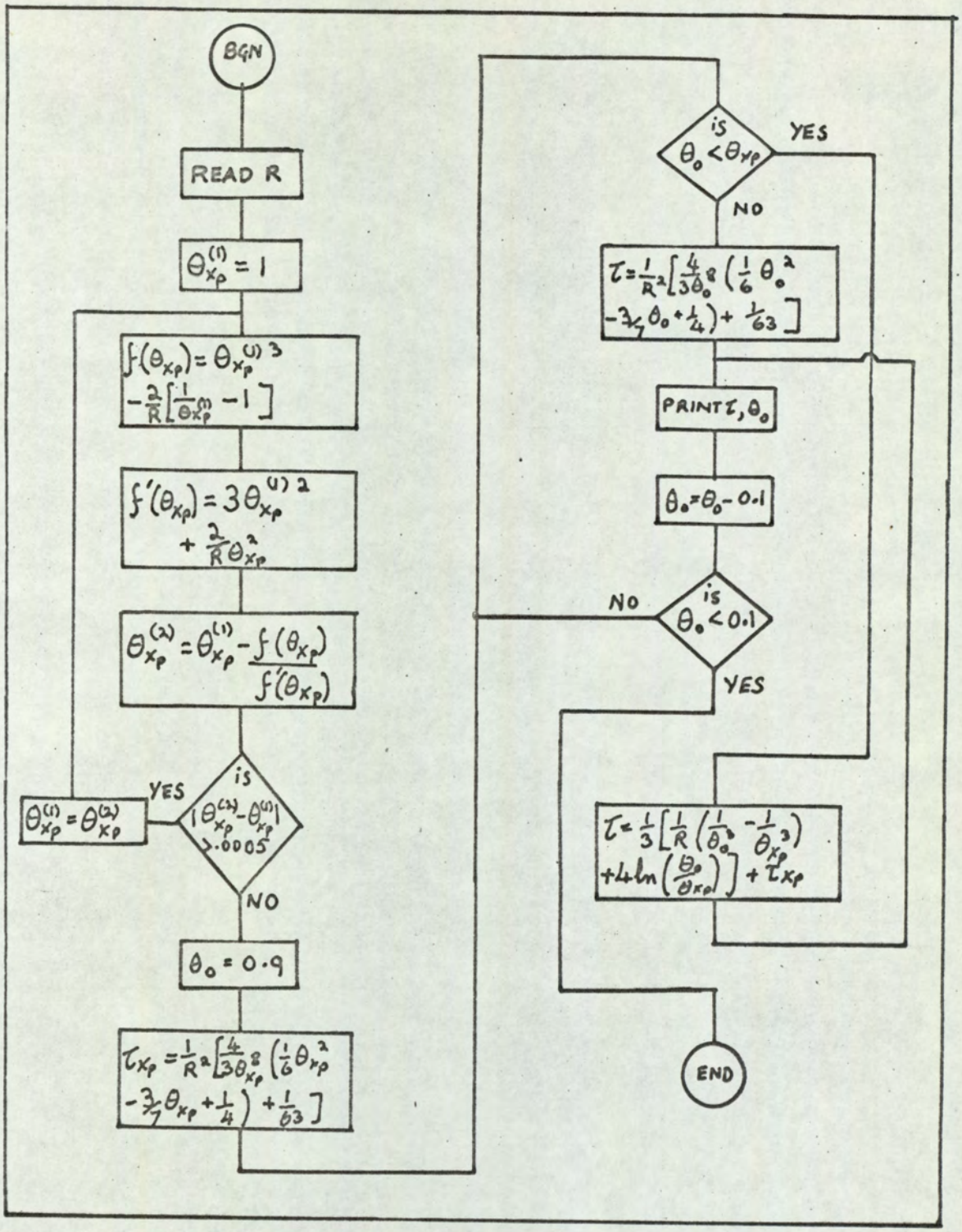


FIG C.3 FLOW SHEET FOR INTEGRAL METHOD WITH RADIATION BOUNDARY CONDITION.

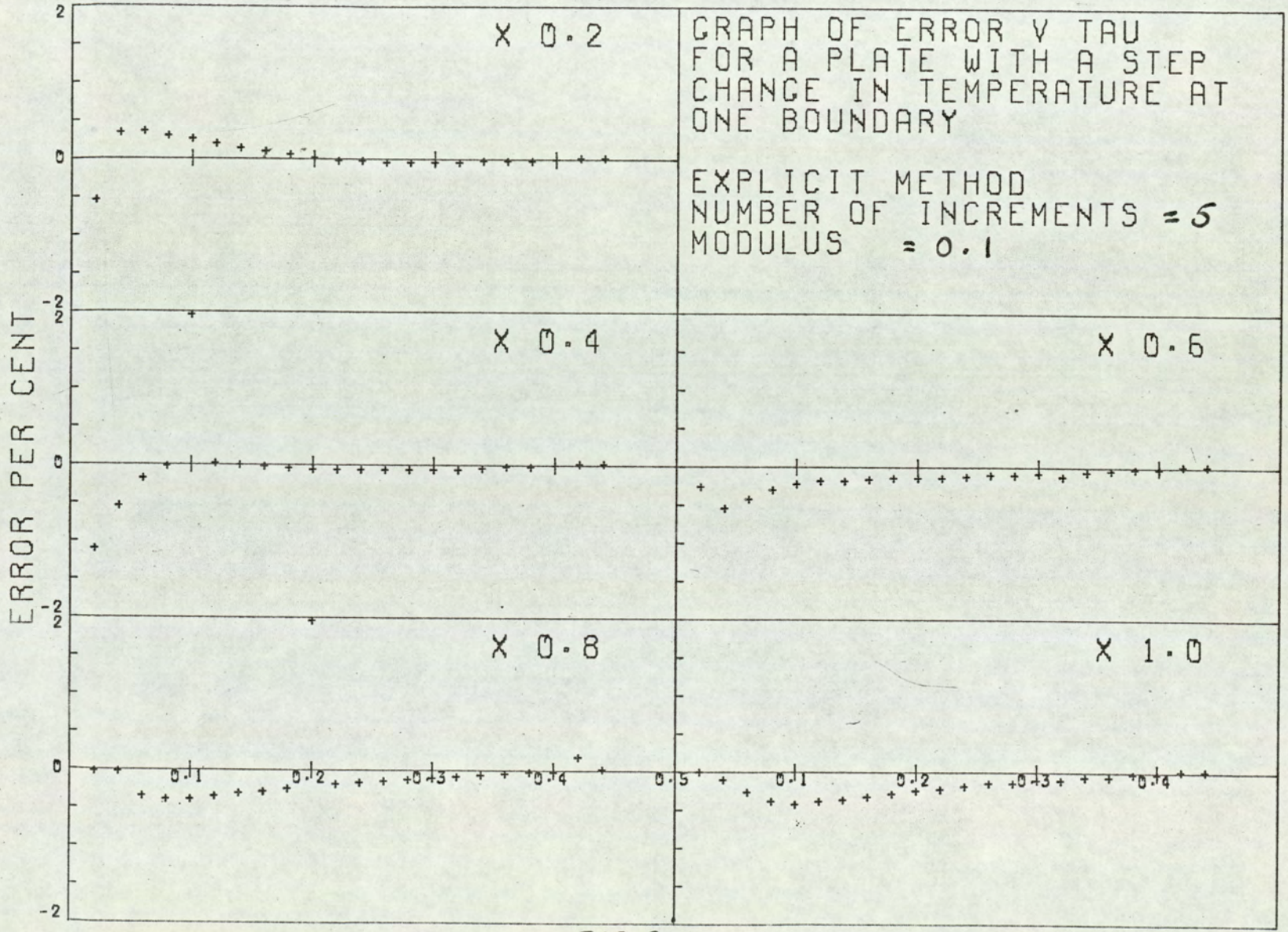

GRAPH OF ERROR V TAU
FOR A PLATE WITH A STEP
CHANGE IN TEMPERATURE AT
ONE BOUNDARY
EXPLICIT METHOD
NUMBER OF INCREMENTS = 5
MODULUS = 0.1
X 0.2
X 0.4
X 0.6
X 0.8
X 1.0
ERROR PER CENT
2
0
-2
0.1
0.2
0.3
0.4
0.5

FIG C.4

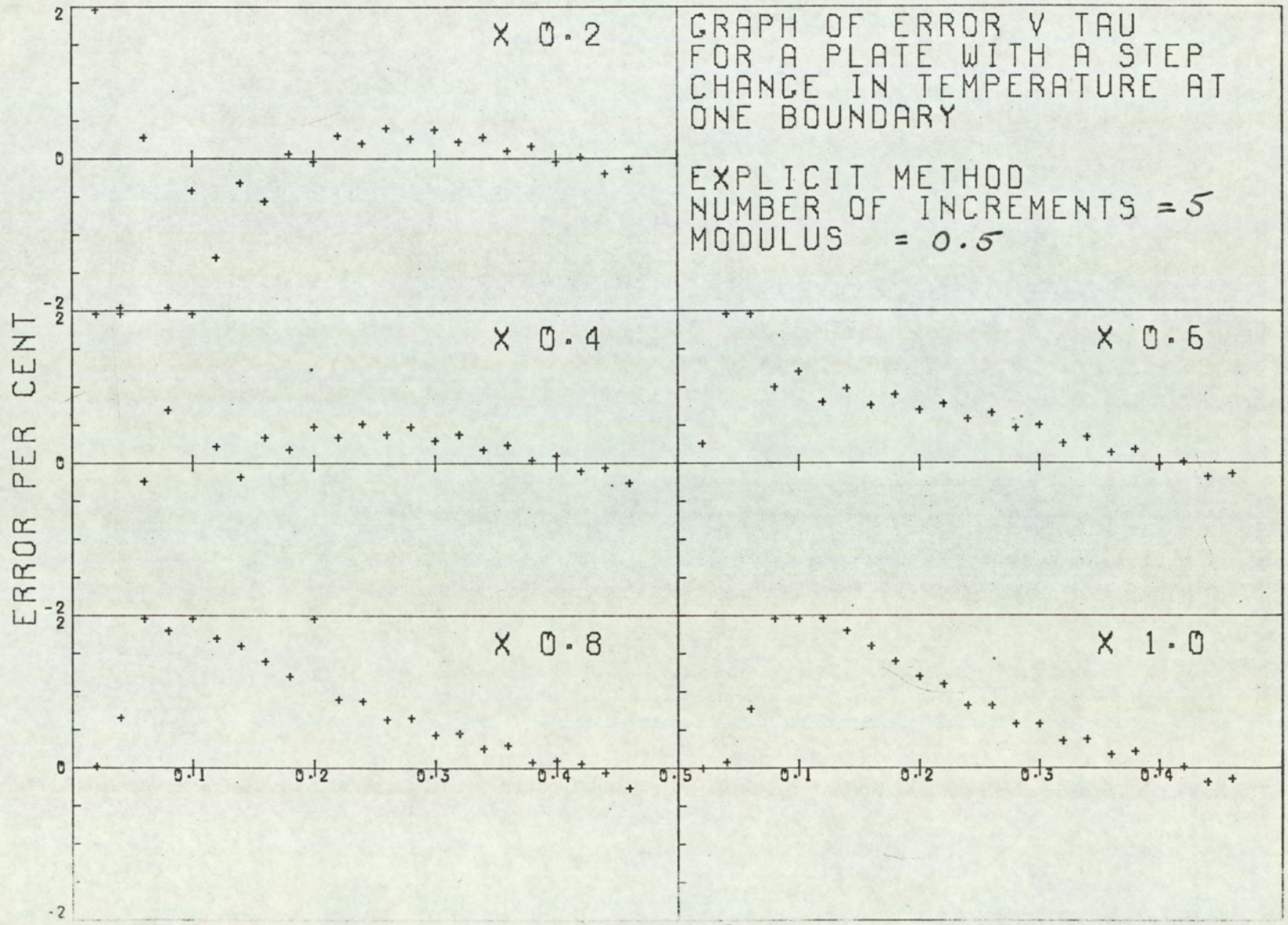

GRAPH OF ERROR V TAU
FOR A PLATE WITH A STEP
CHANGE IN TEMPERATURE AT
ONE BOUNDARY
EXPLICIT METHOD
NUMBER OF INCREMENTS = 5
MODULUS = 0.5
X 0.2
X 0.4
X 0.6
X 0.8
X 1.0
ERROR PER CENT
2
0
-2
0.1
0.2
0.3
0.4
0.5

FIG.5

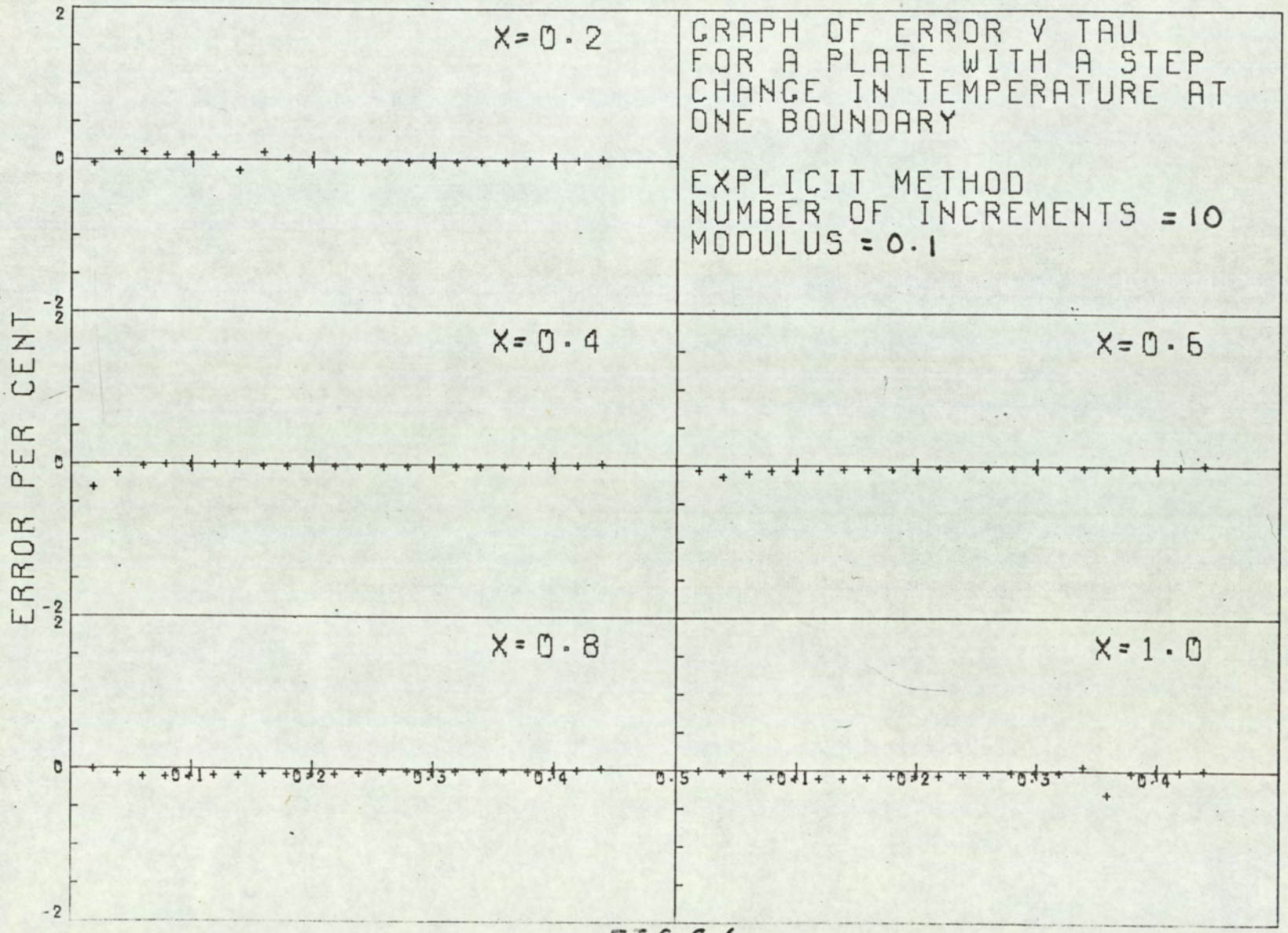

X=0.2
X=0.4
X=0.6
X=0.8
X=1.0
GRAPH OF ERROR V TAU
FOR A PLATE WITH A STEP
CHANGE IN TEMPERATURE AT
ONE BOUNDARY
EXPLICIT METHOD
NUMBER OF INCREMENTS = 10
MODULUS = 0.1
ERROR PER CENT
2
0
-2
0.1
0.2
0.3
0.4
0.5

FIG C.6

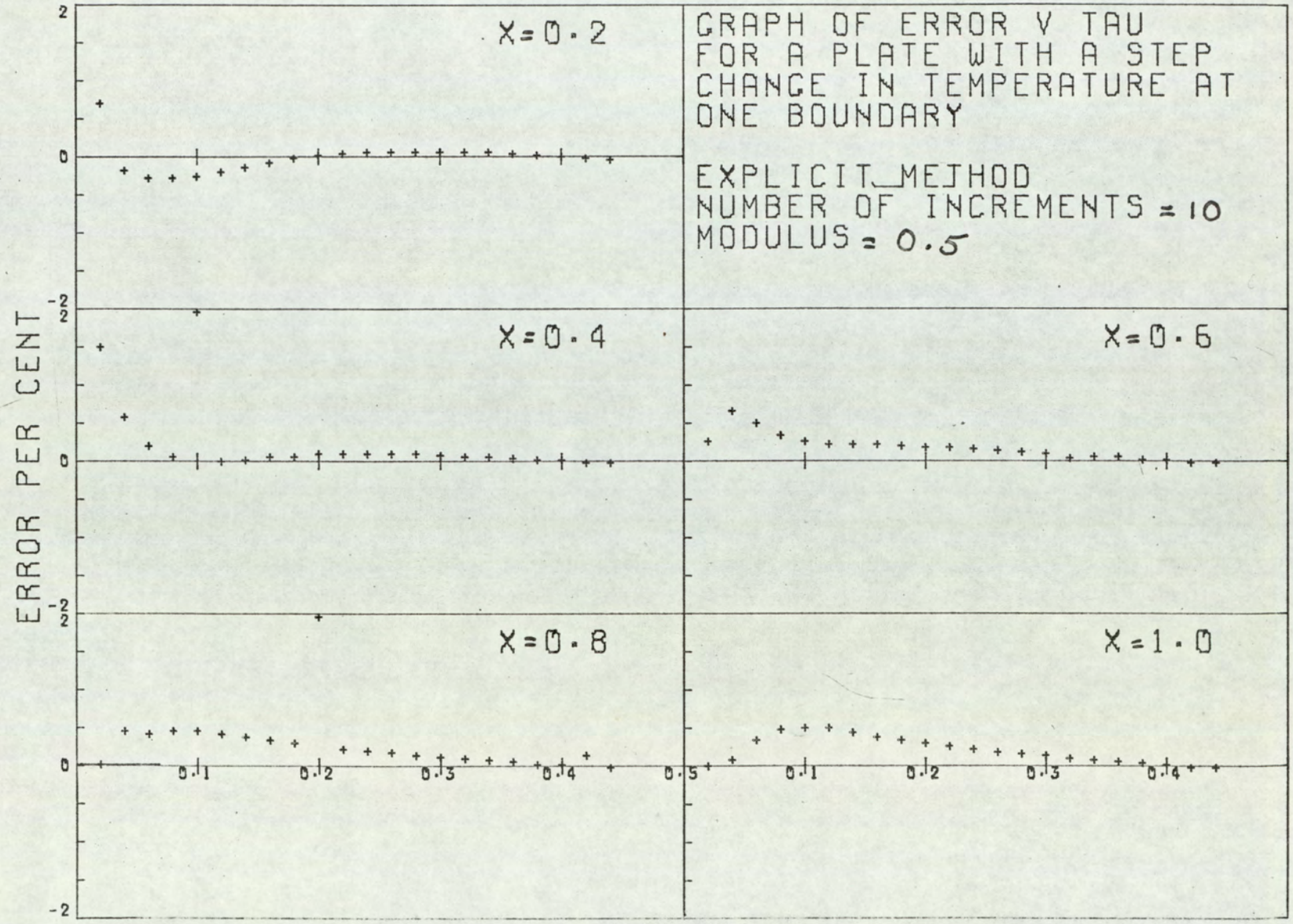

GRAPH OF ERROR V TAU
FOR A PLATE WITH A STEP
CHANGE IN TEMPERATURE AT
ONE BOUNDARY
EXPLICIT METHOD
NUMBER OF INCREMENTS = 10
MODULUS = 0.5
X = 0.2
X = 0.4
X = 0.6
X = 0.8
X = 1.0
ERROR PER CENT
2
0
-2
0.1
0.2
0.3
0.4
0.5

FIG C.7

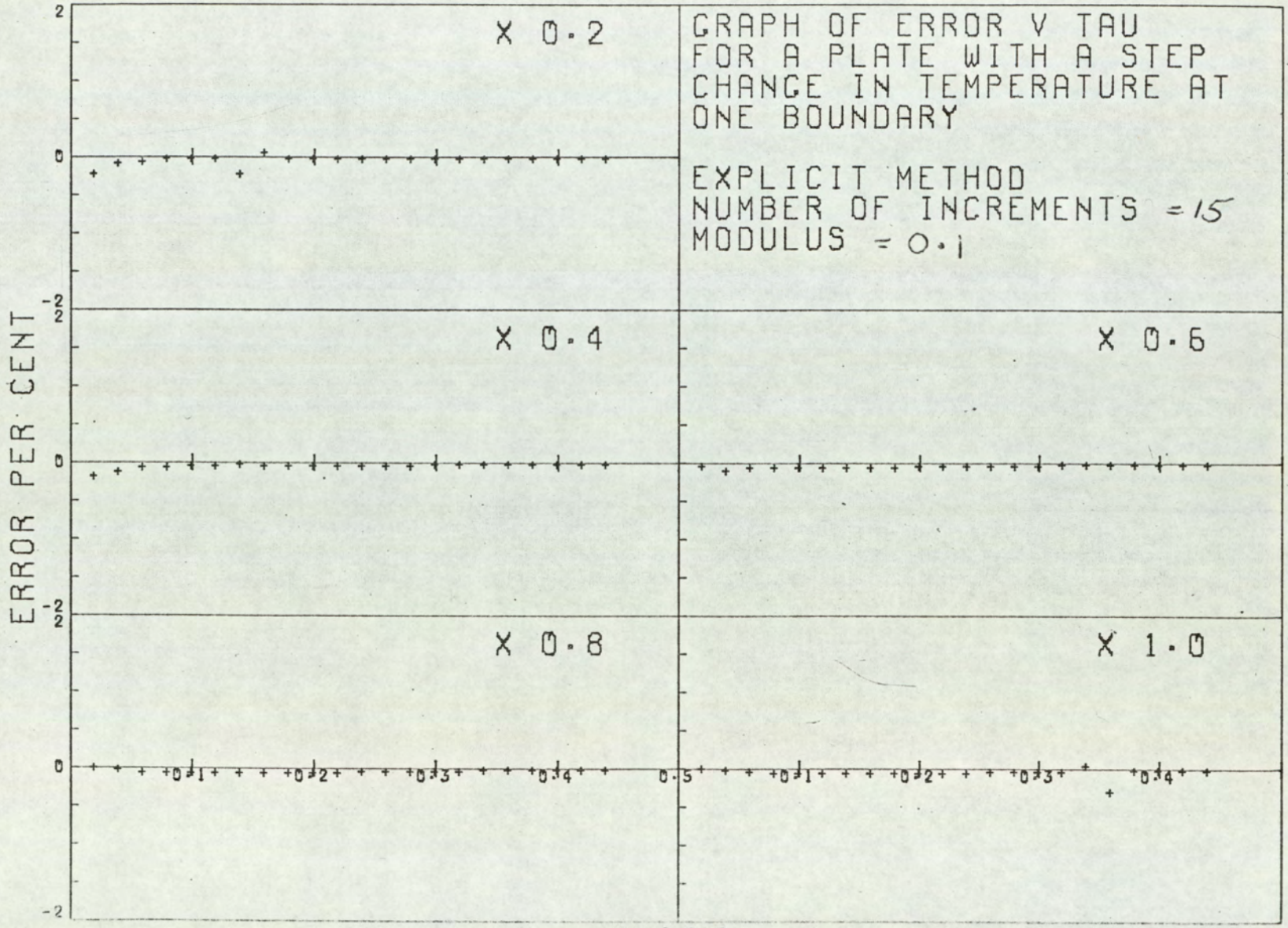

GRAPH OF ERROR V TAU
FOR A PLATE WITH A STEP
CHANGE IN TEMPERATURE AT
ONE BOUNDARY
EXPLICIT METHOD
NUMBER OF INCREMENTS = 15
MODULUS = 0.1
X 0.2
X 0.4
X 0.6
X 0.8
X 1.0
ERROR PER CENT
2
0
-2
0.1
0.2
0.3
0.4
0.5

FIG.C.8

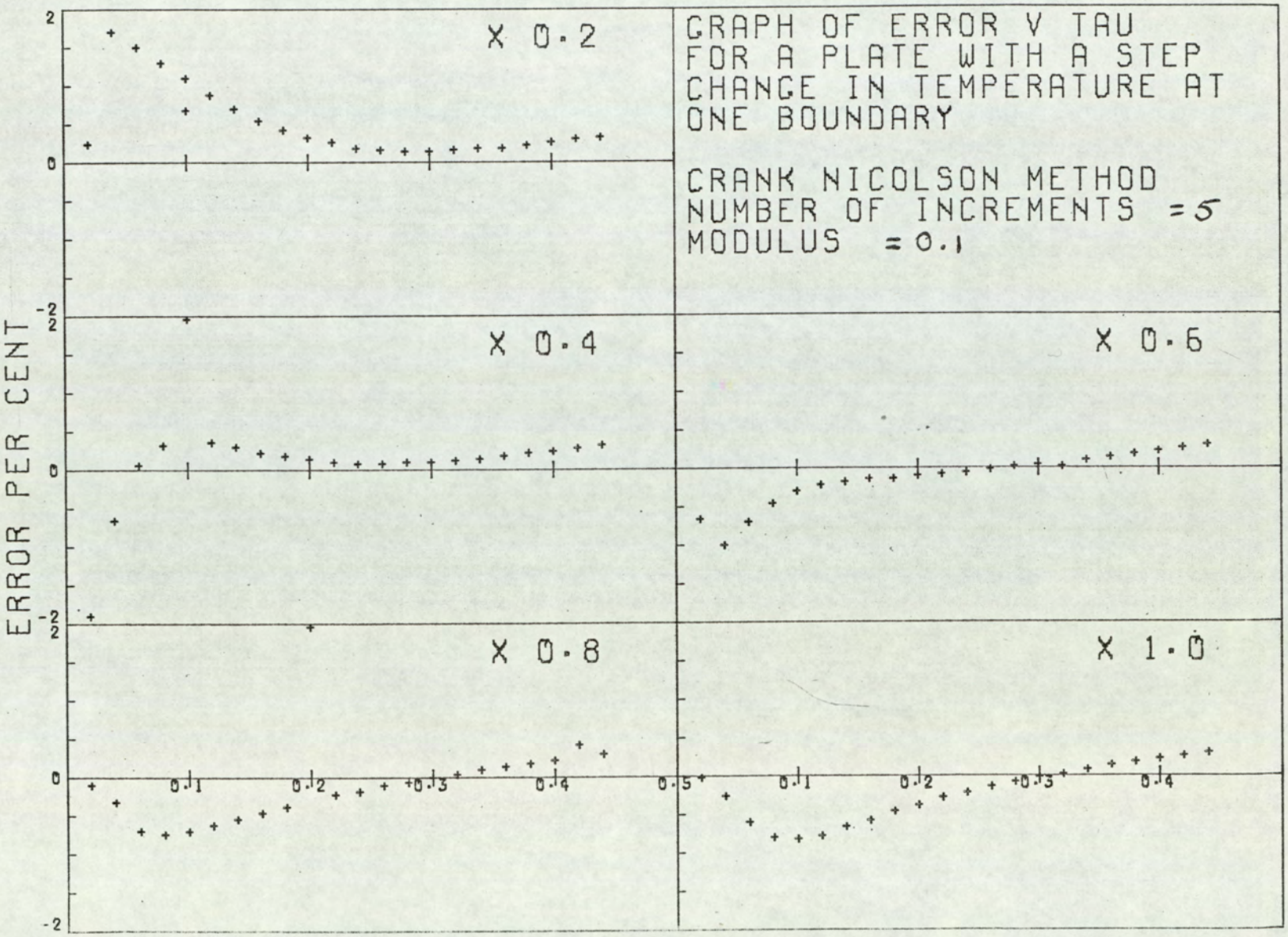

GRAPH OF ERROR V TAU
FOR A PLATE WITH A STEP
CHANGE IN TEMPERATURE AT
ONE BOUNDARY
CRANK NICOLSON METHOD
NUMBER OF INCREMENTS =5
MODULUS =0.1
X 0.2
X 0.4
X 0.6
X 0.8
X 1.0
ERROR PER CENT
2
0
-2
0.1
0.2
0.3
0.4
0.5

FIG C.9

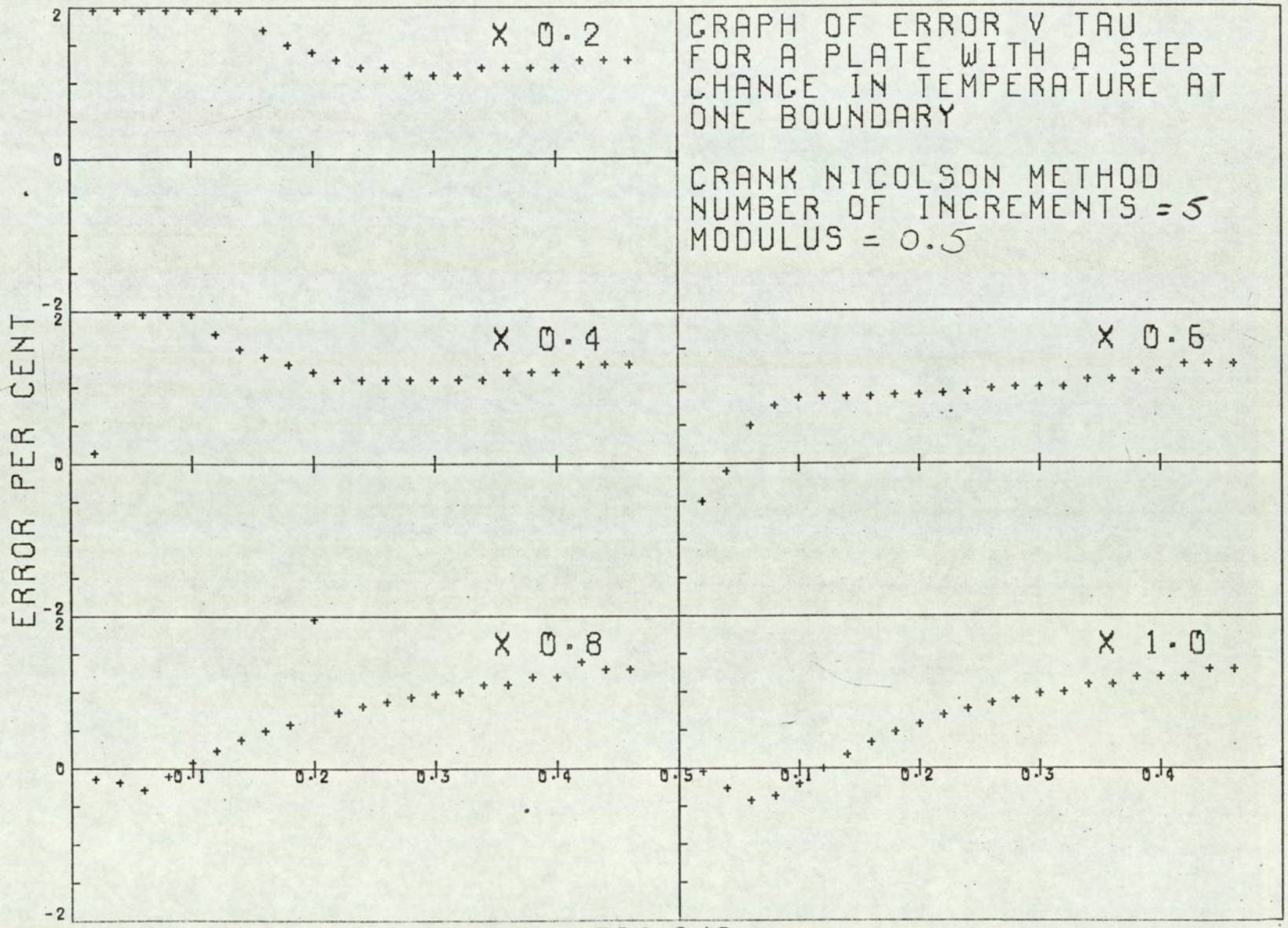

GRAPH OF ERROR V TAU
FOR A PLATE WITH A STEP
CHANGE IN TEMPERATURE AT
ONE BOUNDARY
CRANK NICOLSON METHOD
NUMBER OF INCREMENTS = 5
MODULUS = 0.5
X 0·2
X 0·4
X 0·6
X 0·8
X 1·0
ERROR PER CENT
2
0
-2
0·1
0·2
0·3
0·4
0·5

FIG C.10

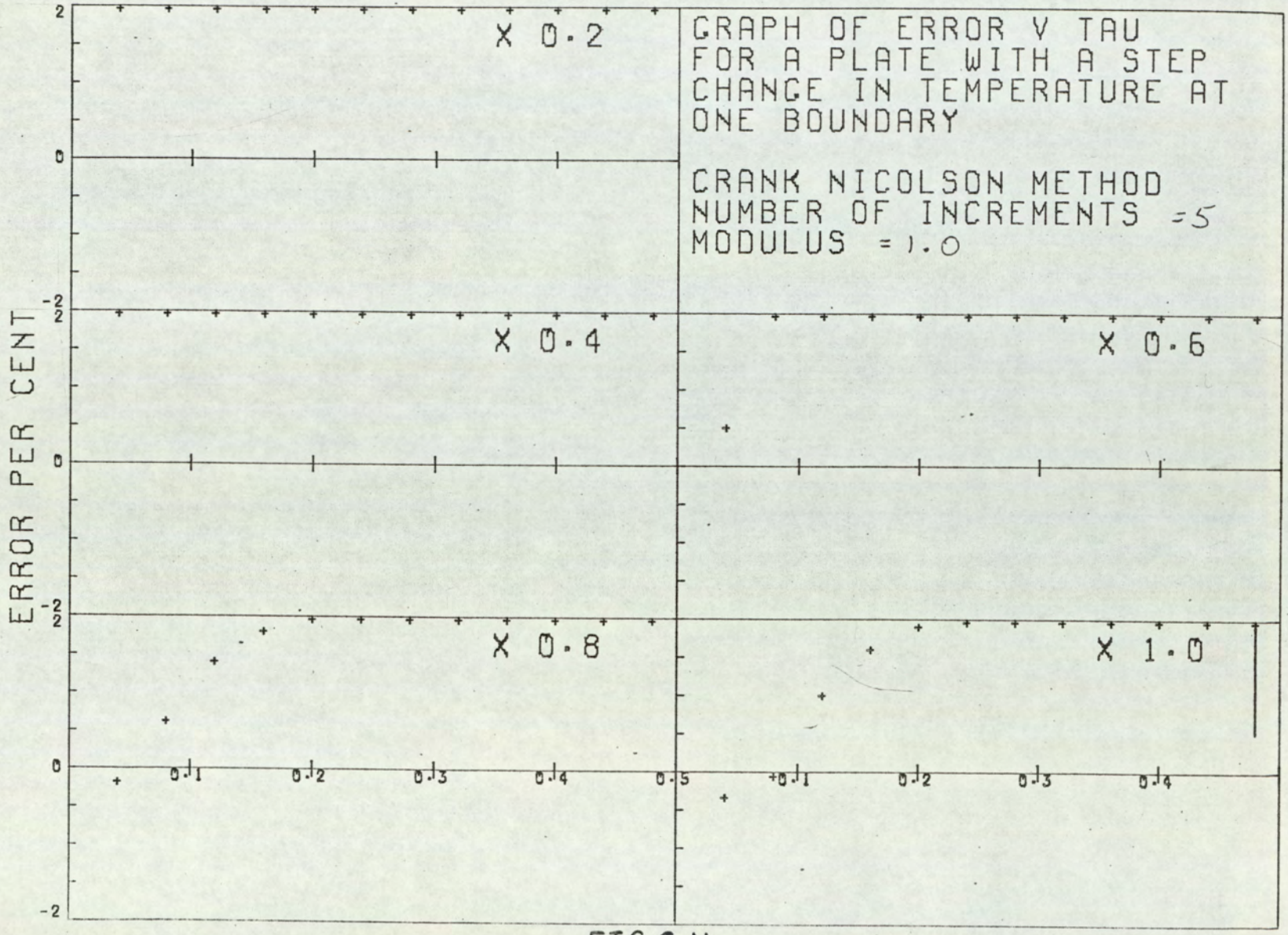
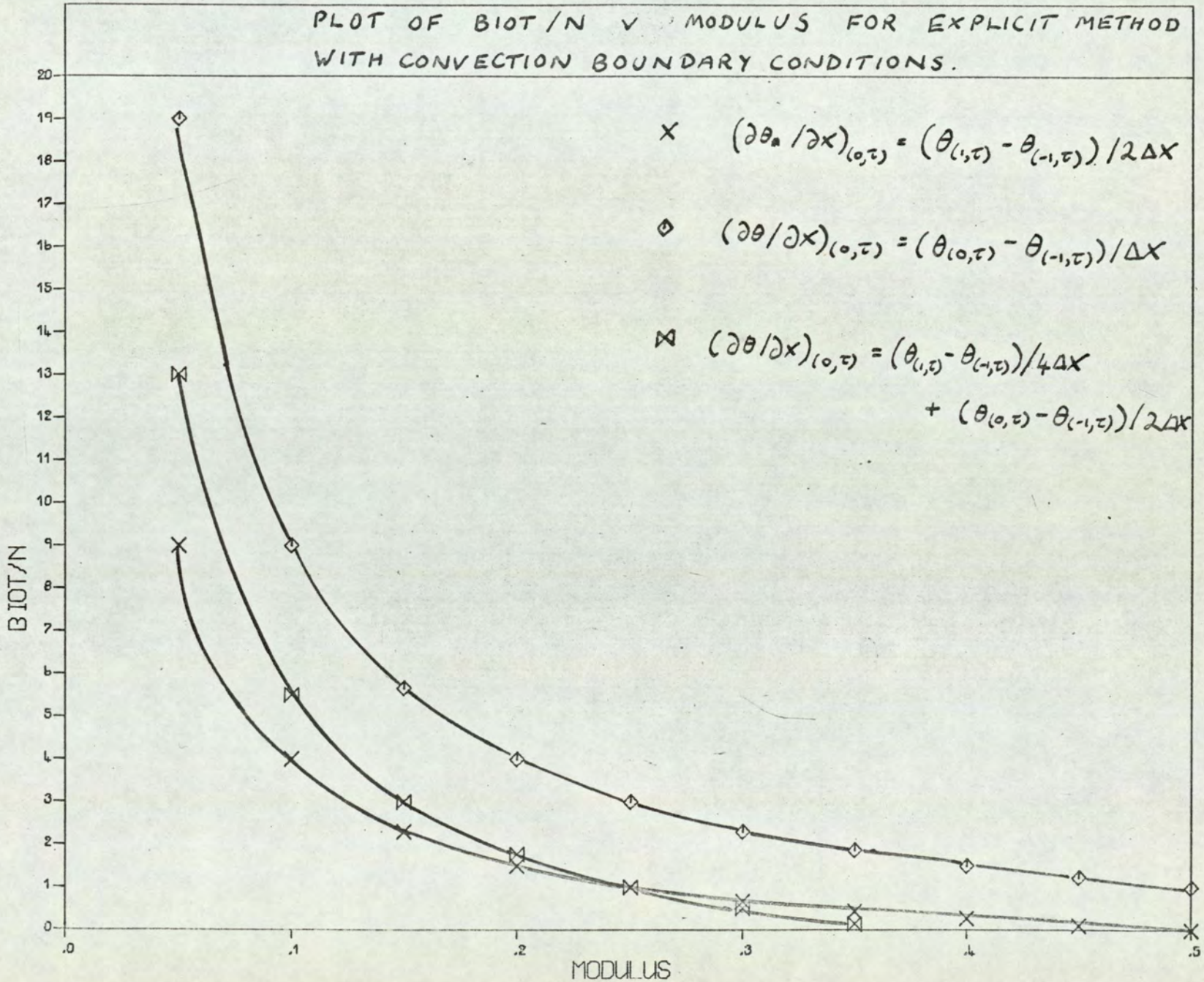

GRAPH OF ERROR V TAU
FOR A PLATE WITH A STEP
CHANGE IN TEMPERATURE AT
ONE BOUNDARY
CRANK NICOLSON METHOD
NUMBER OF INCREMENTS =5
MODULUS = 1.0
X 0.2
X 0.4
X 0.6
X 0.8
X 1.0
ERROR PER CENT
2
0
-2
0.1
0.2
0.3
0.4
0.5

FIG C.11


PLOT OF BIOT/N V MODULUS FOR EXPLICIT METHOD WITH CONVECTION BOUNDARY CONDITIONS.
× $(\partial\theta_n/\partial x)_{(0,\tau)} = (\theta_{(1,\tau)} - \theta_{(-1,\tau)})/2\Delta x$
◇ $(\partial\theta/\partial x)_{(0,\tau)} = (\theta_{(0,\tau)} - \theta_{(-1,\tau)})/\Delta x$
⋈ $(\partial\theta/\partial x)_{(0,\tau)} = (\theta_{(1,\tau)} - \theta_{(-1,\tau)})/4\Delta x + (\theta_{(0,\tau)} - \theta_{(-1,\tau)})/2\Delta x$
BIOT/N
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
.0
.1
.2
.3
.4
.5
MODULUS
FIG 6.12

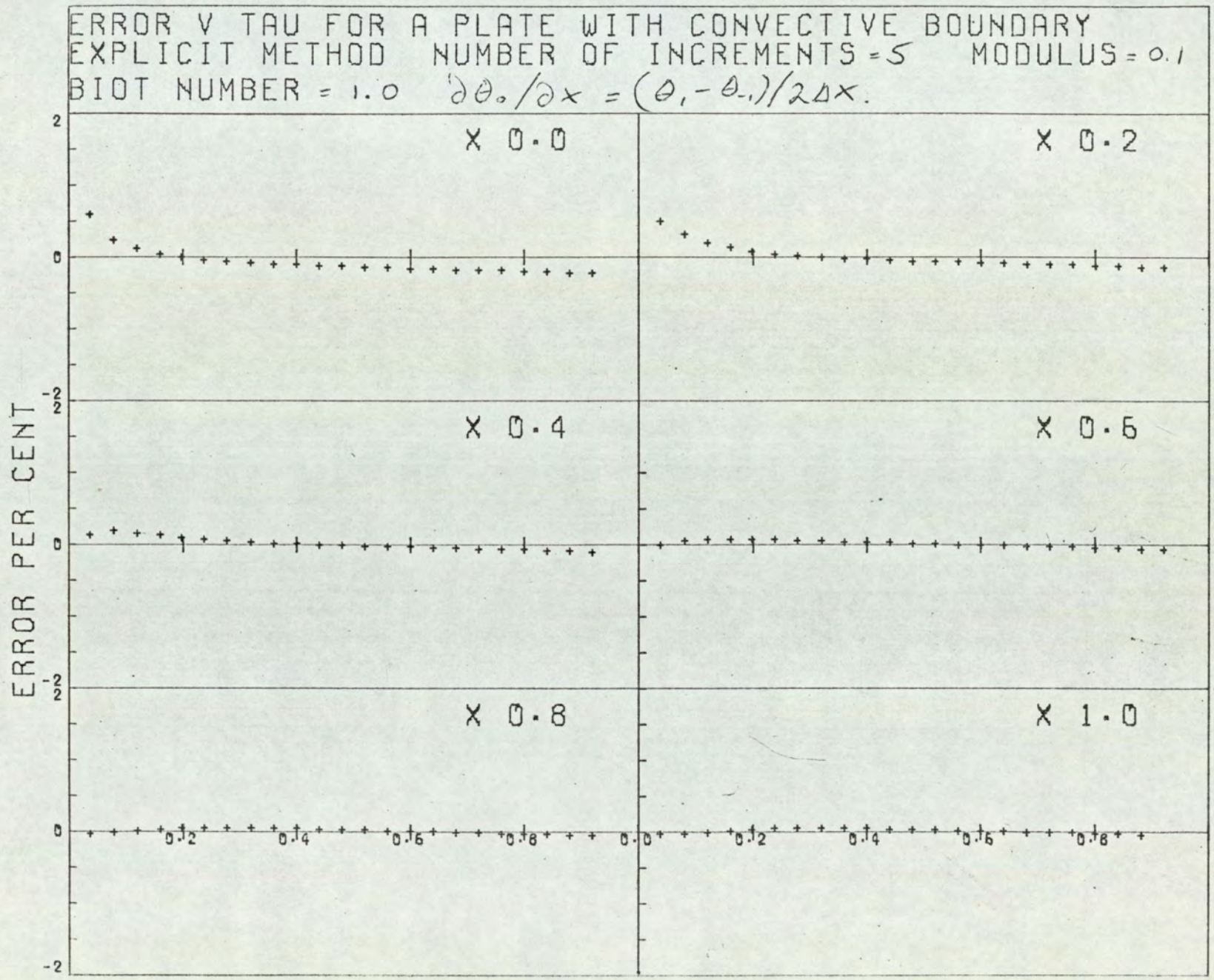

ERROR V TAU FOR A PLATE WITH CONVECTIVE BOUNDARY
EXPLICIT METHOD NUMBER OF INCREMENTS = 5 MODULUS = 0.1
BIOT NUMBER = 1.0 $\partial\theta_0/\partial x = (\theta_1 - \theta_{-1})/2\Delta x$.
X 0.0
X 0.2
X 0.4
X 0.6
X 0.8
X 1.0
ERROR PER CENT
2
0
-2
0.2
0.4
0.6
0.8
1.0
TAU
TAU

FIG C.13

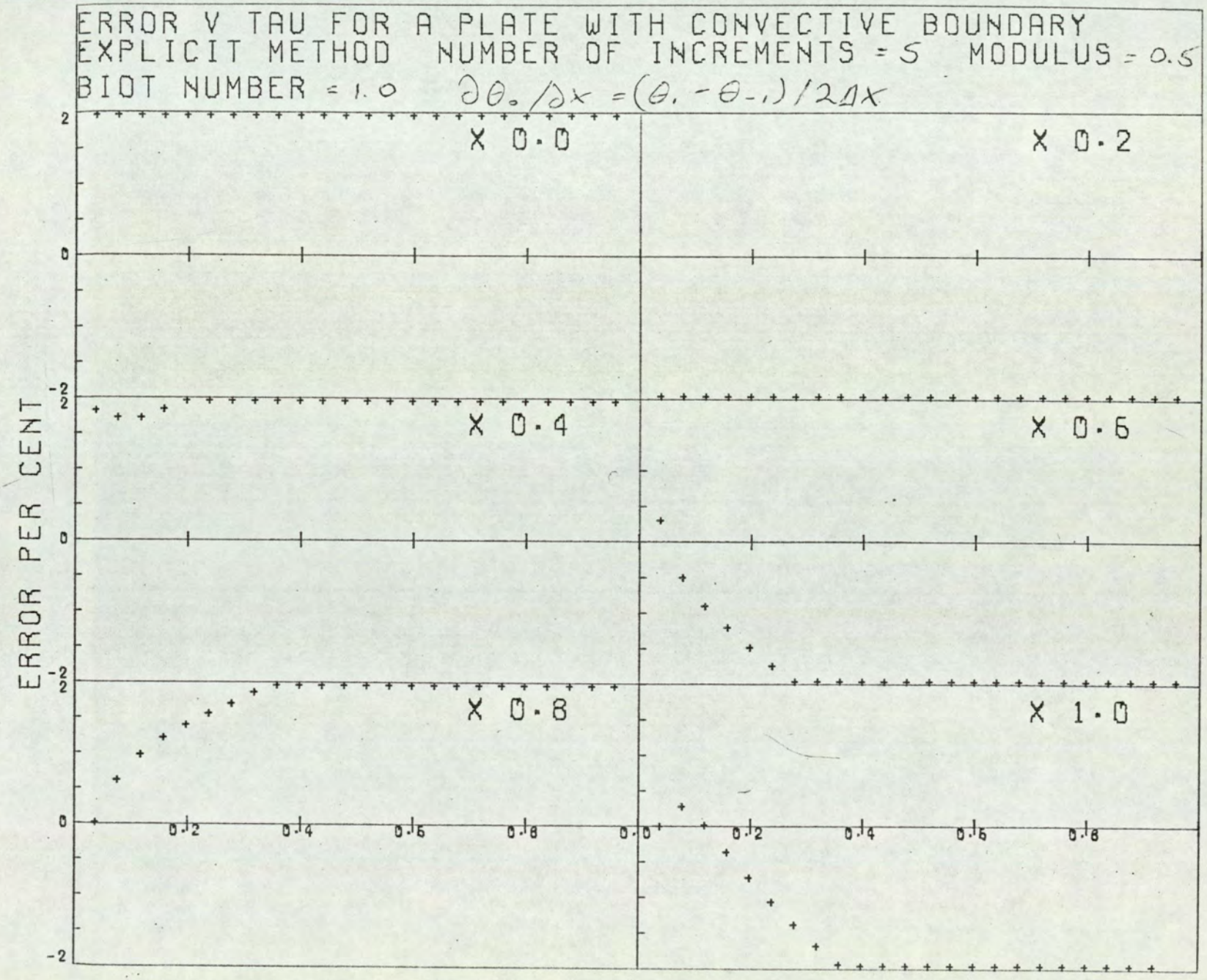

ERROR V TAU FOR A PLATE WITH CONVECTIVE BOUNDARY
EXPLICIT METHOD NUMBER OF INCREMENTS = 5 MODULUS = 0.5
BIOT NUMBER = 1.0 ∂θ0/∂x = (θ1 - θ-1)/2Δx
X 0.0
X 0.2
X 0.4
X 0.6
X 0.8
X 1.0
ERROR PER CENT
TAU
TAU

FIG C.14

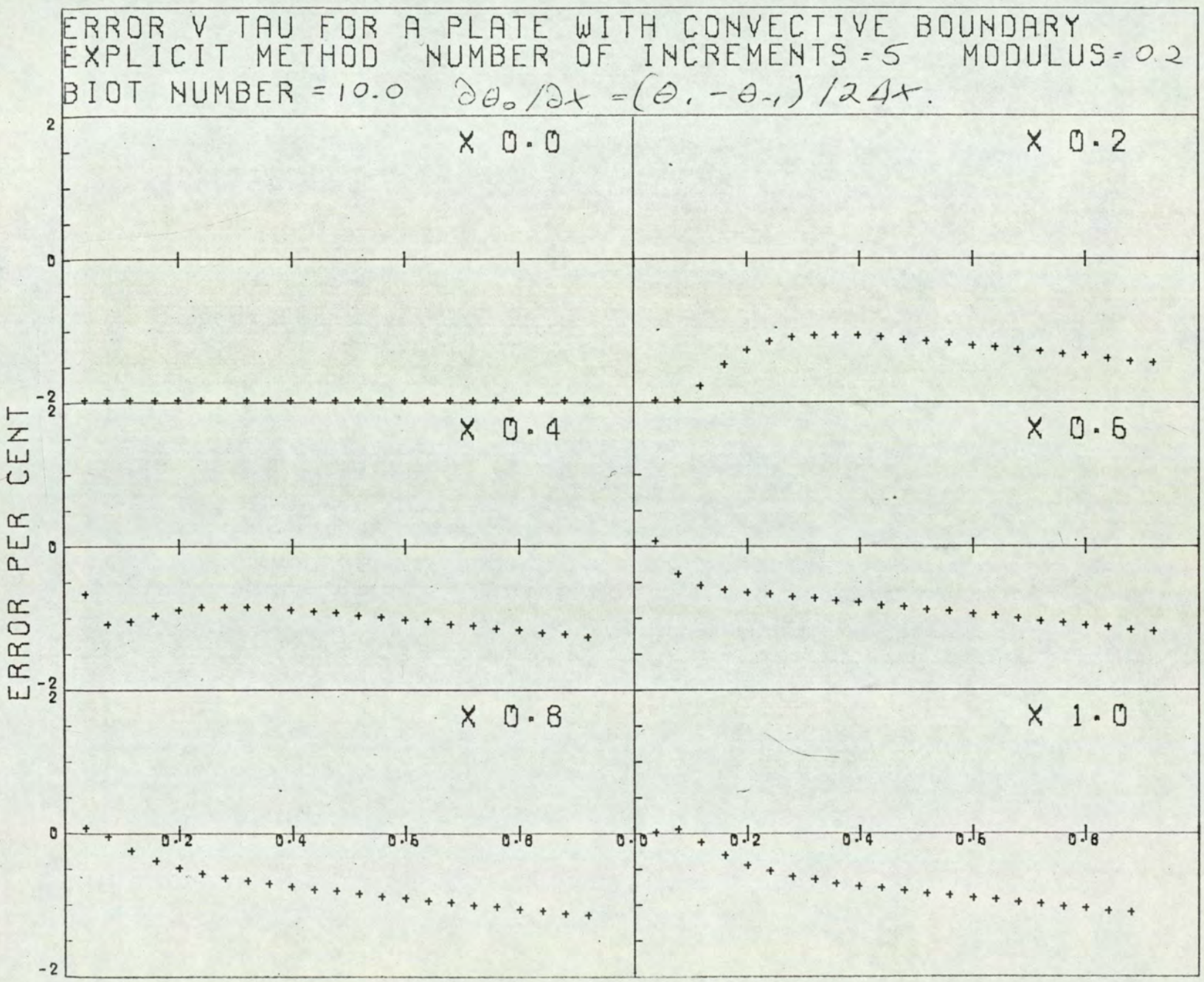

ERROR V TAU FOR A PLATE WITH CONVECTIVE BOUNDARY
EXPLICIT METHOD NUMBER OF INCREMENTS = 5 MODULUS = 0.2
BIOT NUMBER = 10.0 $\partial\theta_0/\partial x = (\theta_1 - \theta_{-1})/2\Delta x$.
X 0.0
X 0.2
X 0.4
X 0.6
X 0.8
X 1.0
ERROR PER CENT
2
0
-2
0.0
0.2
0.4
0.6
0.8
TAU

FIG C.15

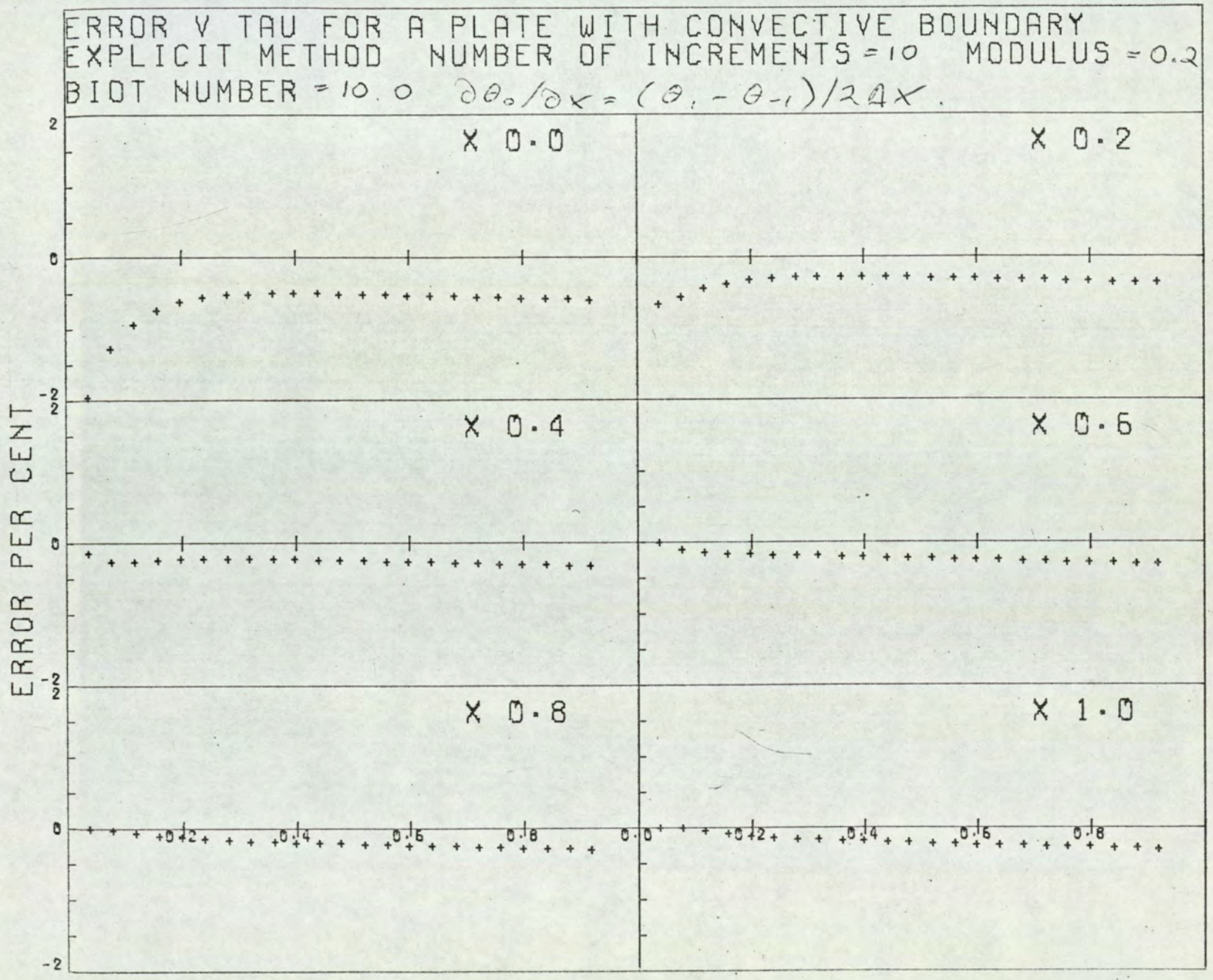

ERROR V TAU FOR A PLATE WITH CONVECTIVE BOUNDARY
EXPLICIT METHOD NUMBER OF INCREMENTS = 10 MODULUS = 0.2
BIOT NUMBER = 10.0 ∂θ0/∂x = (θ1 - θ-1)/2Δx.
X 0.0
X 0.2
X 0.4
X 0.6
X 0.8
X 1.0
ERROR PER CENT
2
0
-2
0.2
0.4
0.6
0.8
1.0
TAU

FIG C.16

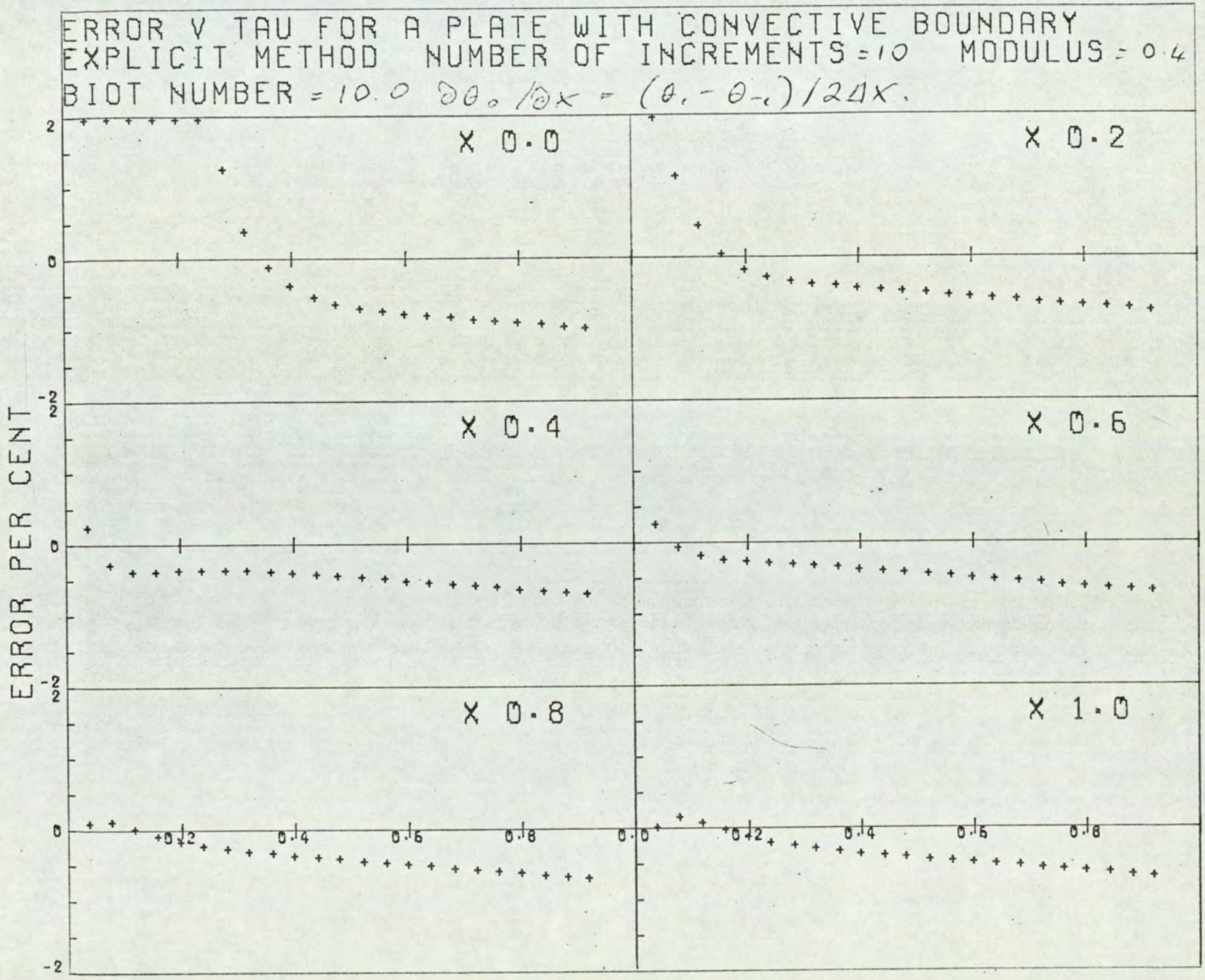

ERROR V TAU FOR A PLATE WITH CONVECTIVE BOUNDARY
EXPLICIT METHOD NUMBER OF INCREMENTS = 10 MODULUS = 0.4
BIOT NUMBER = 10.0 ∂θ₀/∂x = (θ₁ - θ₋₁)/2Δx.
X 0.0
X 0.2
X 0.4
X 0.6
X 0.8
X 1.0
ERROR PER CENT
2
0
-2
0.0
0.2
0.4
0.6
0.8
TAU

FIG C.17

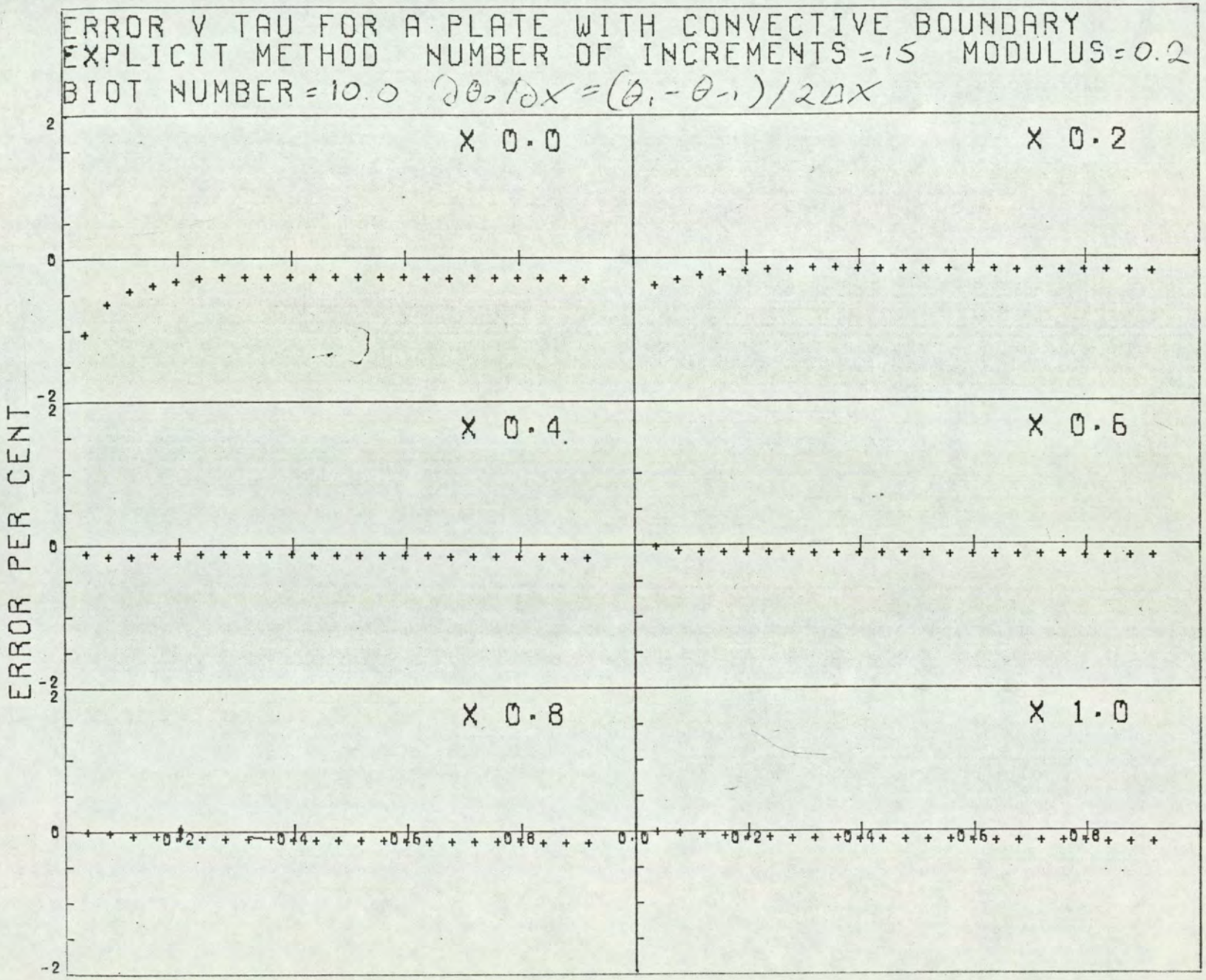

ERROR V TAU FOR A PLATE WITH CONVECTIVE BOUNDARY
EXPLICIT METHOD NUMBER OF INCREMENTS = 15 MODULUS = 0.2
BIOT NUMBER = 10.0 ∂θ/∂x = (θ1 - θ-1)/2Δx
X 0.0
X 0.2
X 0.4
X 0.6
X 0.8
X 1.0
ERROR PER CENT
2
0
-2
TAU
0.0
0.2
0.4
0.6
0.8

FIG C 18

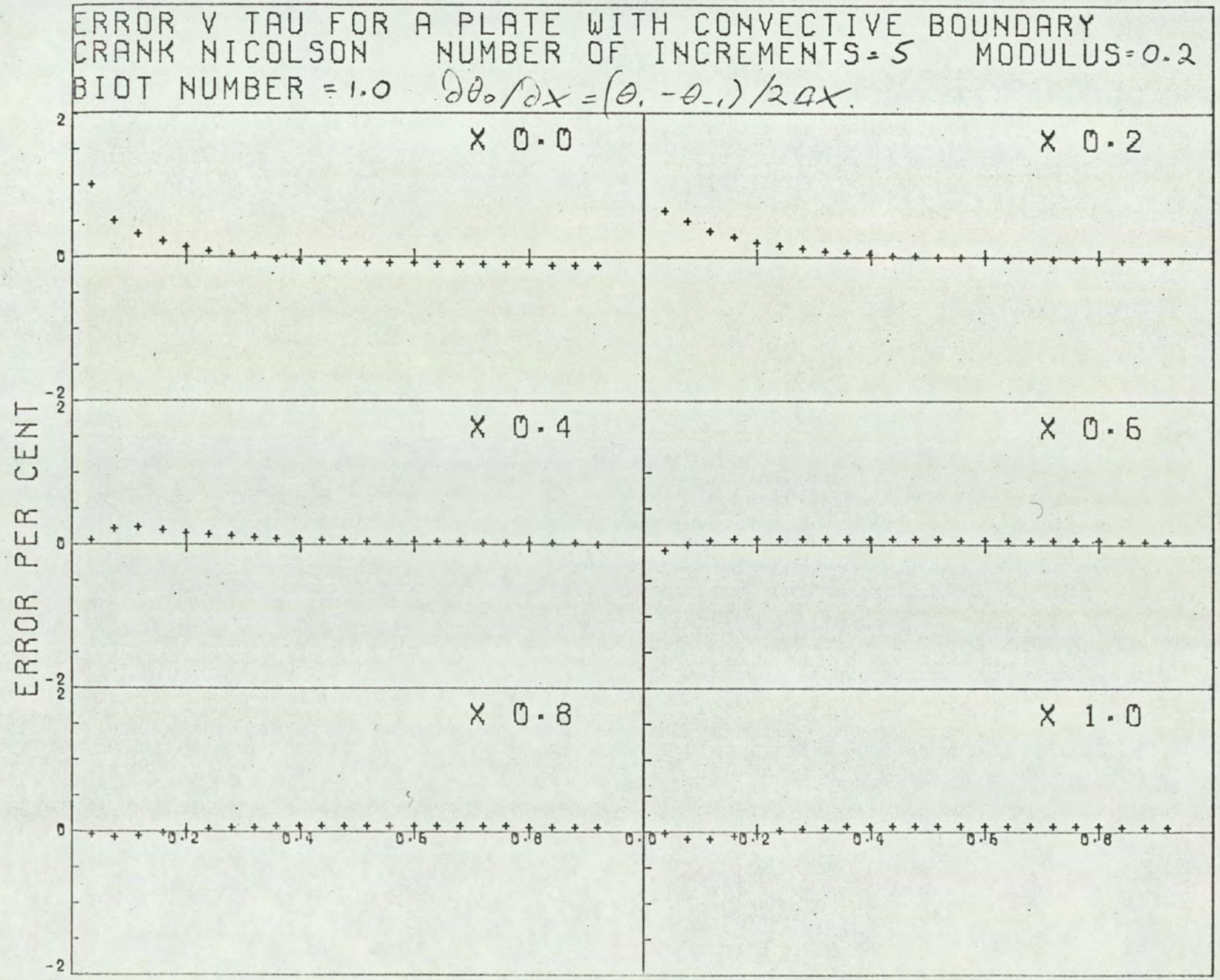

ERROR V TAU FOR A PLATE WITH CONVECTIVE BOUNDARY
CRANK NICOLSON NUMBER OF INCREMENTS=5 MODULUS=0.2
BIOT NUMBER = 1.0 $\partial\theta_0/\partial x = (\theta_1 - \theta_{-1})/2\Delta x$.
X 0.0
X 0.2
X 0.4
X 0.6
X 0.8
X 1.0
ERROR PER CENT
2
0
-2
0.2
0.4
0.6
0.8
TAU

FIG C.19

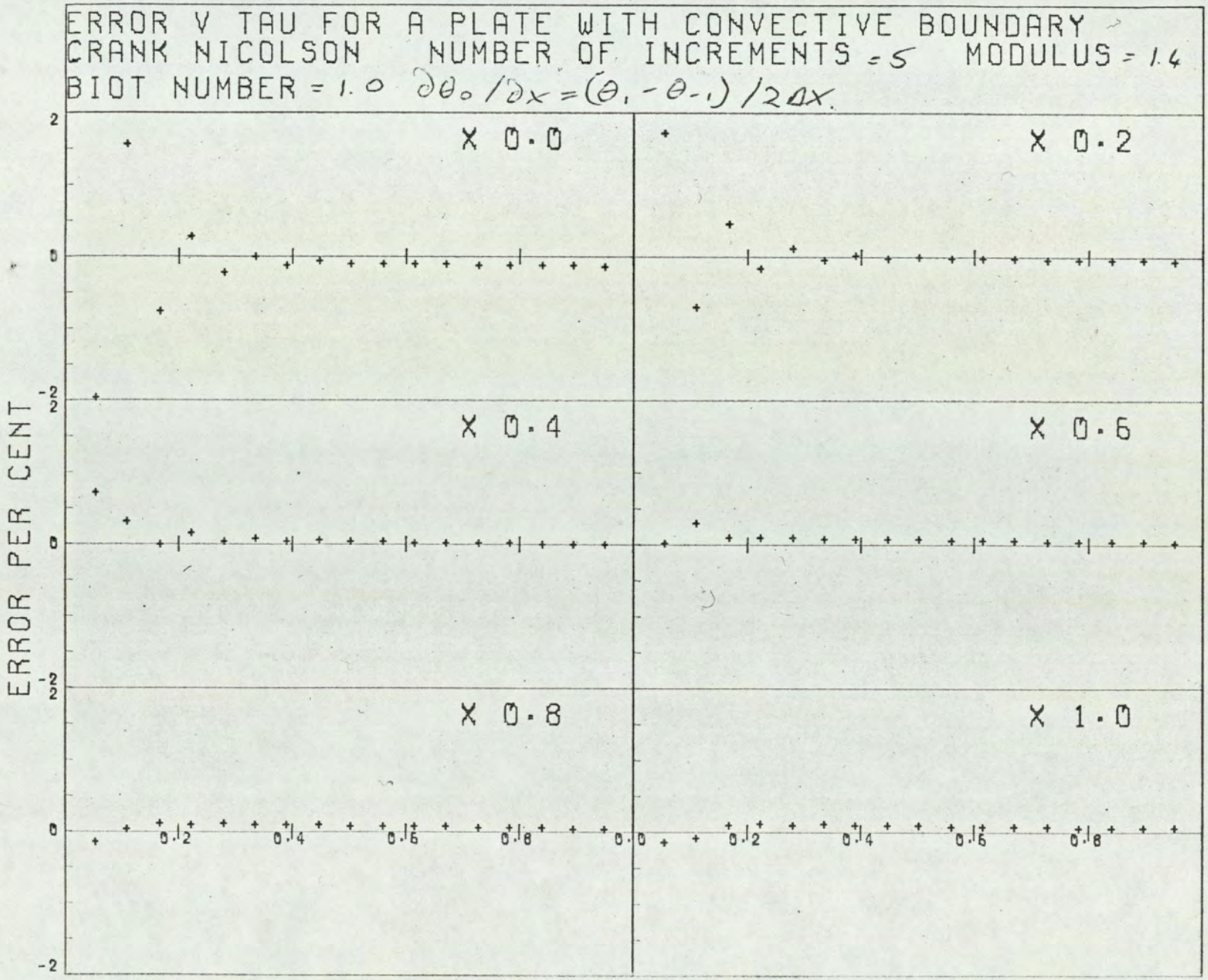

ERROR V TAU FOR A PLATE WITH CONVECTIVE BOUNDARY
CRANK NICOLSON NUMBER OF INCREMENTS = 5 MODULUS = 1.4
BIOT NUMBER = 1.0 ∂θ0/∂x = (θ1 - θ-1)/2Δx.
X 0.0
X 0.2
X 0.4
X 0.6
X 0.8
X 1.0
ERROR PER CENT
2
0
-2
0.12
0.14
0.16
0.18
TAU

FIG 6.20

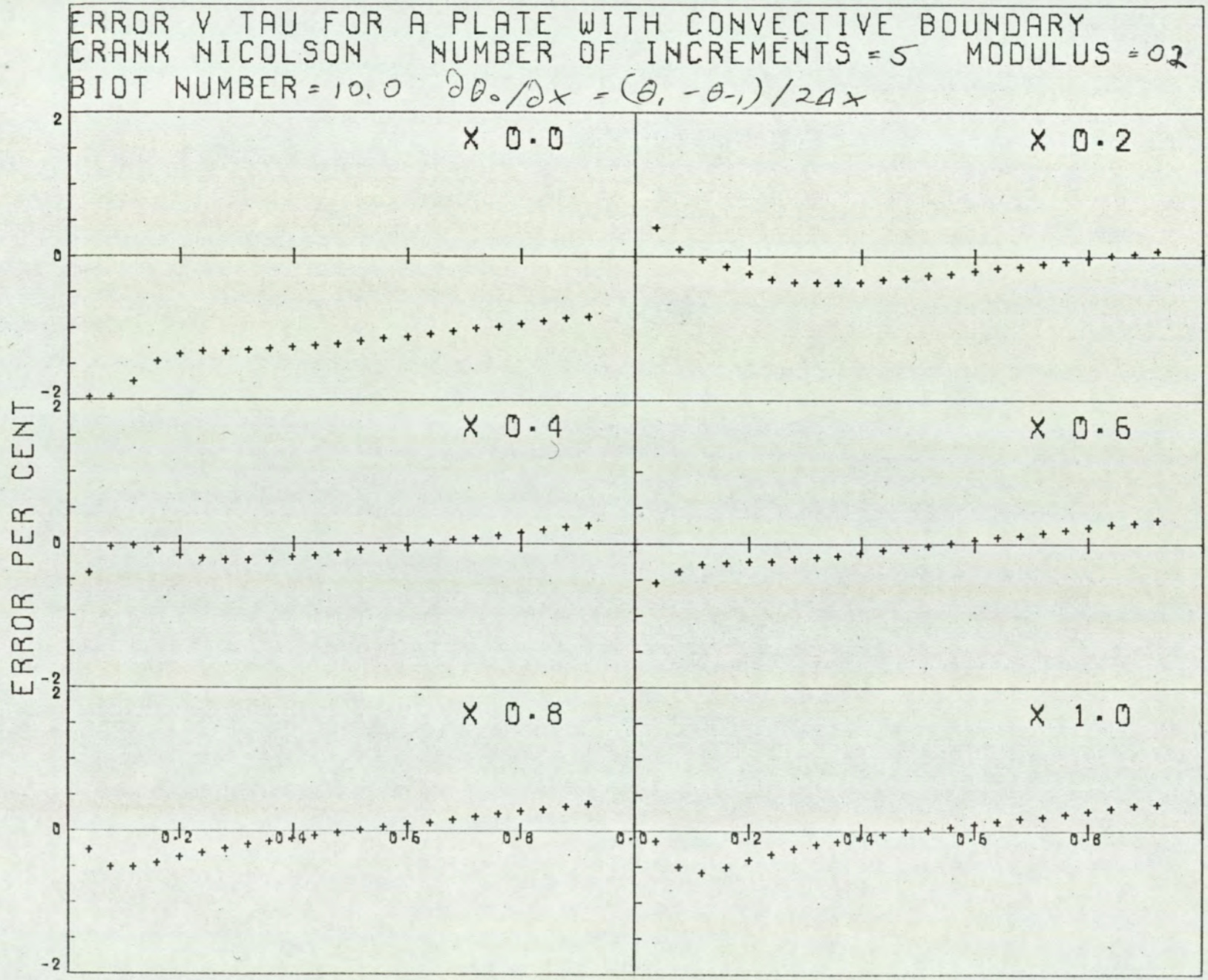

ERROR V TAU FOR A PLATE WITH CONVECTIVE BOUNDARY
CRANK NICOLSON NUMBER OF INCREMENTS = 5 MODULUS = 02
BIOT NUMBER = 10.0 ∂θ0/∂x = (θ1 - θ-1)/2Δx
X 0.0
X 0.2
X 0.4
X 0.6
X 0.8
X 1.0
ERROR PER CENT
2
0
-2
TAU
0.2
0.4
0.6
0.8

FIG 6.01

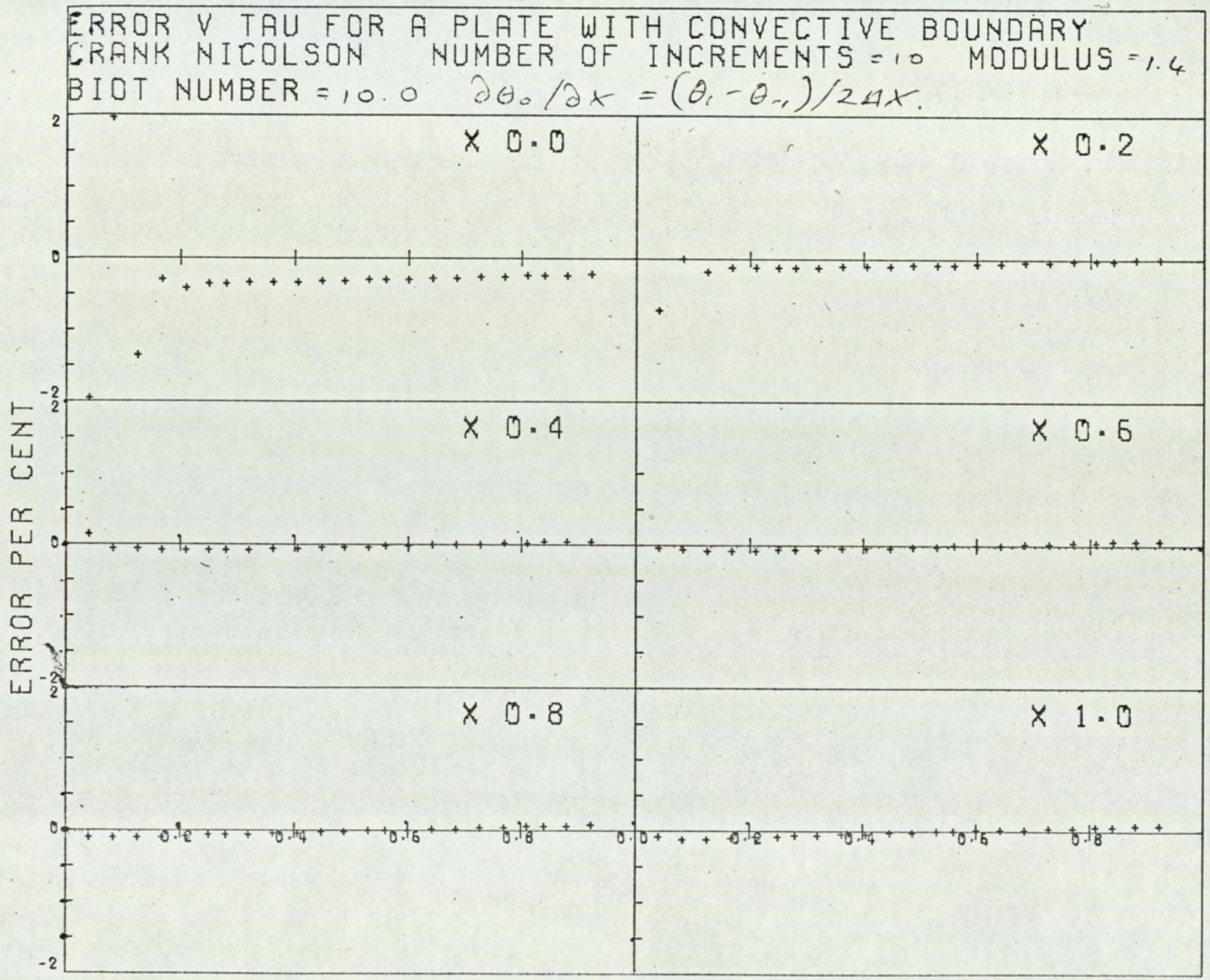

ERROR V TAU FOR A PLATE WITH CONVECTIVE BOUNDARY
CRANK NICOLSON NUMBER OF INCREMENTS = 10 MODULUS = 1.4
BIOT NUMBER = 10.0 ∂θ0/∂x = (θ1 - θ-1)/2Δx.
X 0.0
X 0.2
X 0.4
X 0.6
X 0.8
X 1.0
ERROR PER CENT
2
0
-2
0.2
0.4
0.6
0.8
TAU

FIG C.22

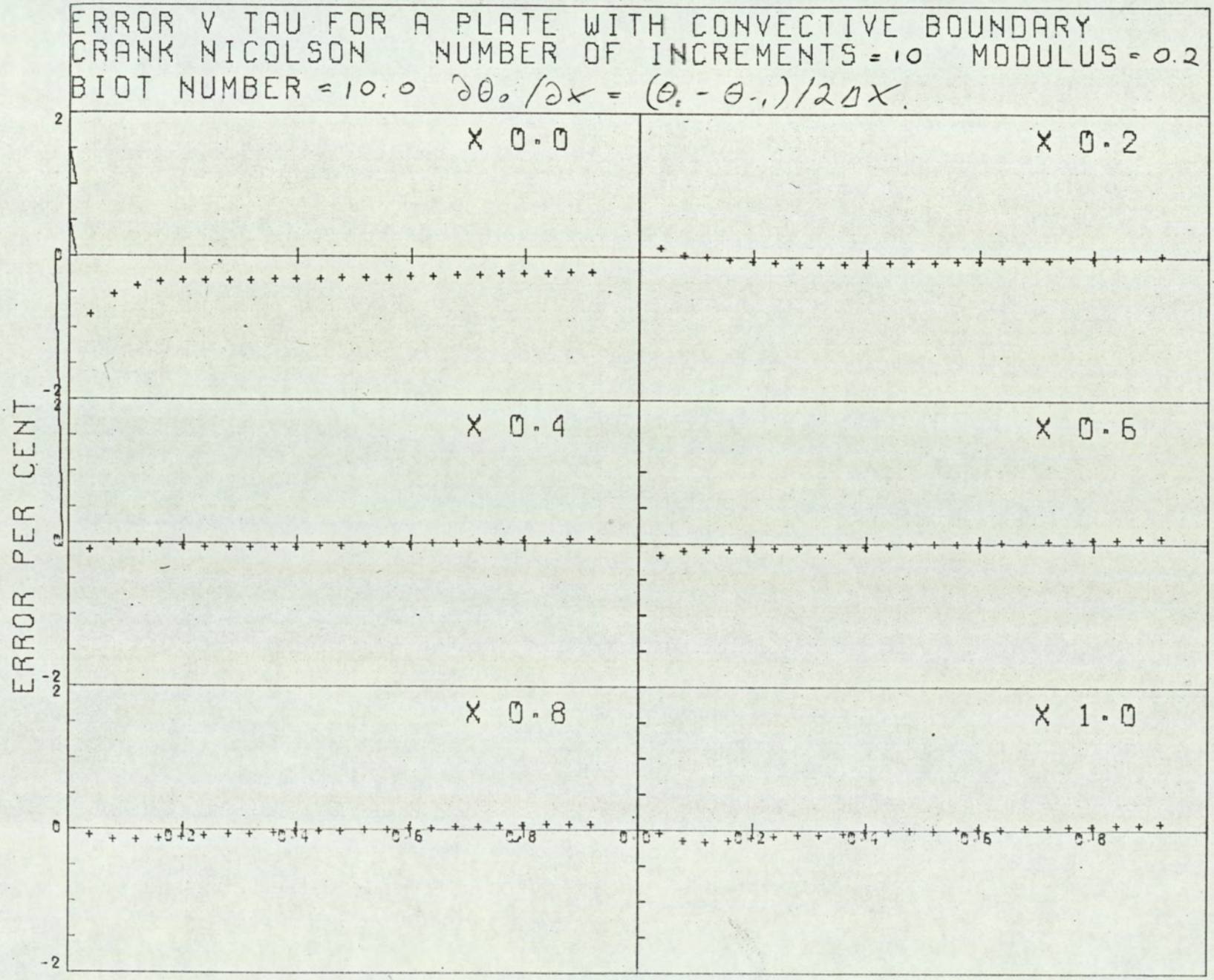

ERROR V TAU FOR A PLATE WITH CONVECTIVE BOUNDARY
CRANK NICOLSON NUMBER OF INCREMENTS = 10 MODULUS = 0.2
BIOT NUMBER = 10.0 ∂θ0/∂x = (θ1 - θ-1)/2Δx.
X 0.0
X 0.2
X 0.4
X 0.6
X 0.8
X 1.0
ERROR PER CENT
2
0
-2
TAU
TAU

FIG C 23

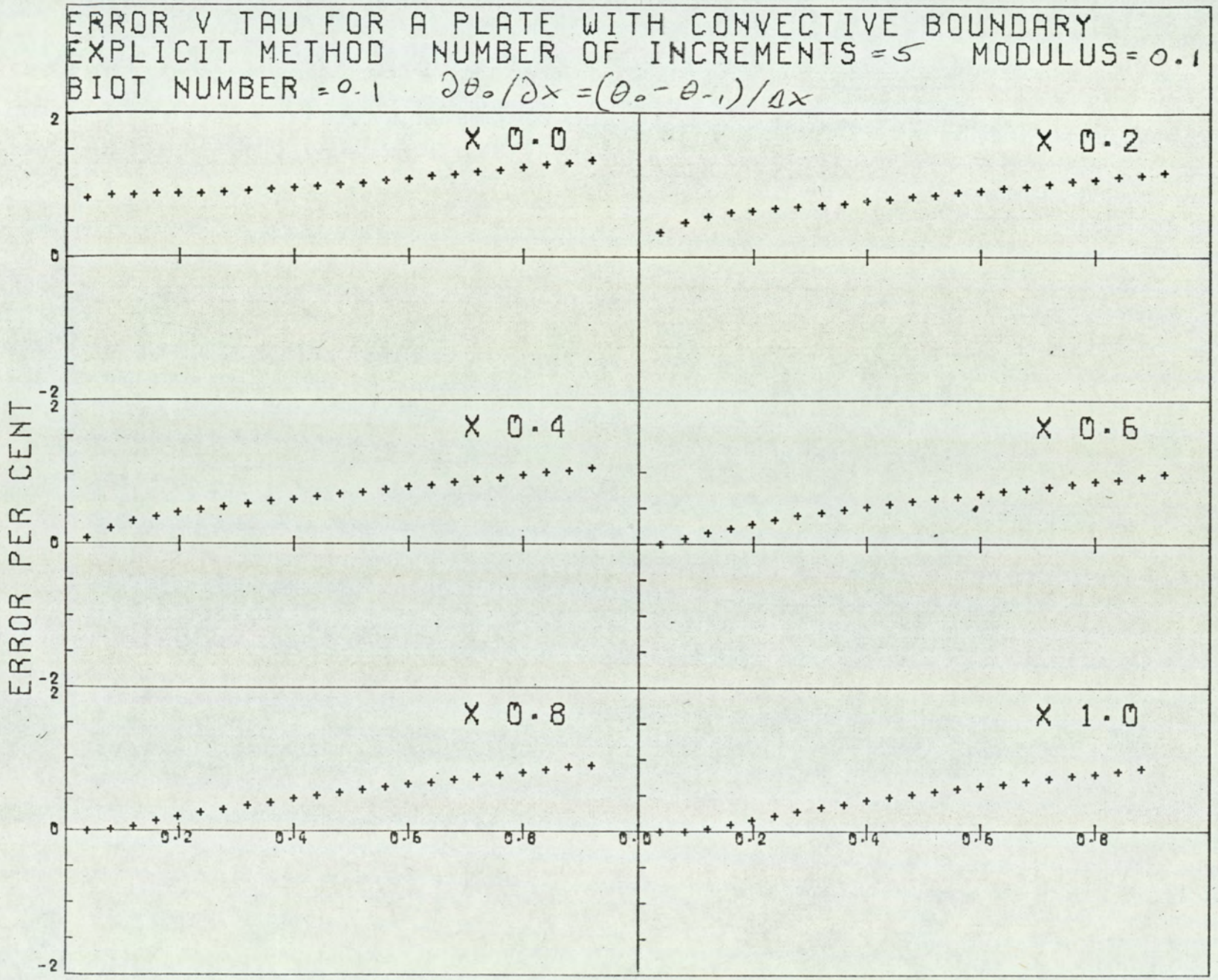

ERROR V TAU FOR A PLATE WITH CONVECTIVE BOUNDARY
EXPLICIT METHOD NUMBER OF INCREMENTS = 5 MODULUS = 0.1
BIOT NUMBER = 0.1 ∂θ0/∂x = (θ0 − θ−1)/Δx
X 0.0
X 0.2
X 0.4
X 0.6
X 0.8
X 1.0
ERROR PER CENT
2
0
-2
0.2
0.4
0.6
0.8
1.0
TAU

FIG C.24

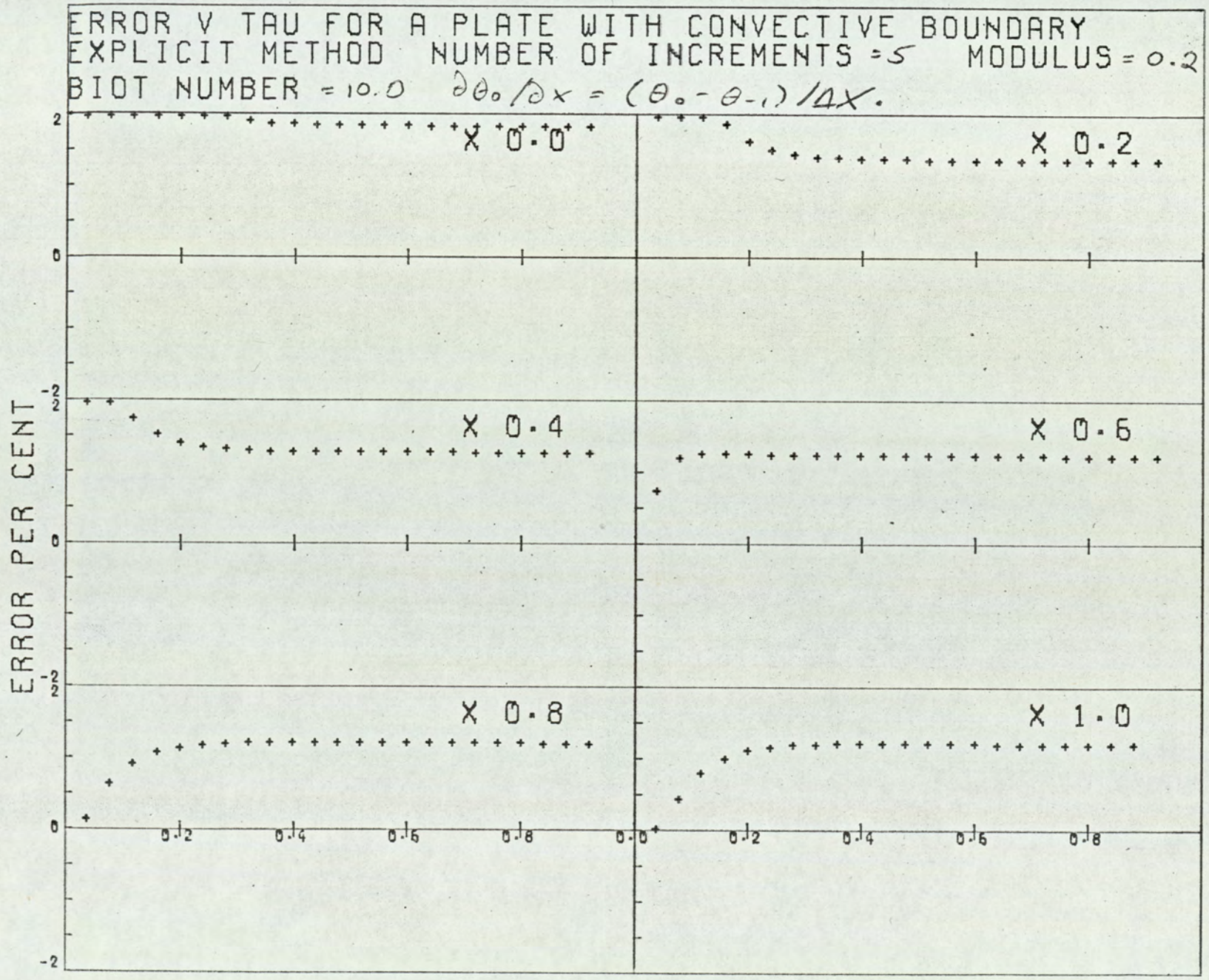

ERROR V TAU FOR A PLATE WITH CONVECTIVE BOUNDARY
EXPLICIT METHOD NUMBER OF INCREMENTS = 5 MODULUS = 0.2
BIOT NUMBER = 10.0 ∂θ0/∂x = (θ0 - θ-1)/Δx.
X 0.0
X 0.2
X 0.4
X 0.6
X 0.8
X 1.0
ERROR PER CENT
2
0
-2
0.2
0.4
0.6
0.8
1.0
TAU

FIG C.25

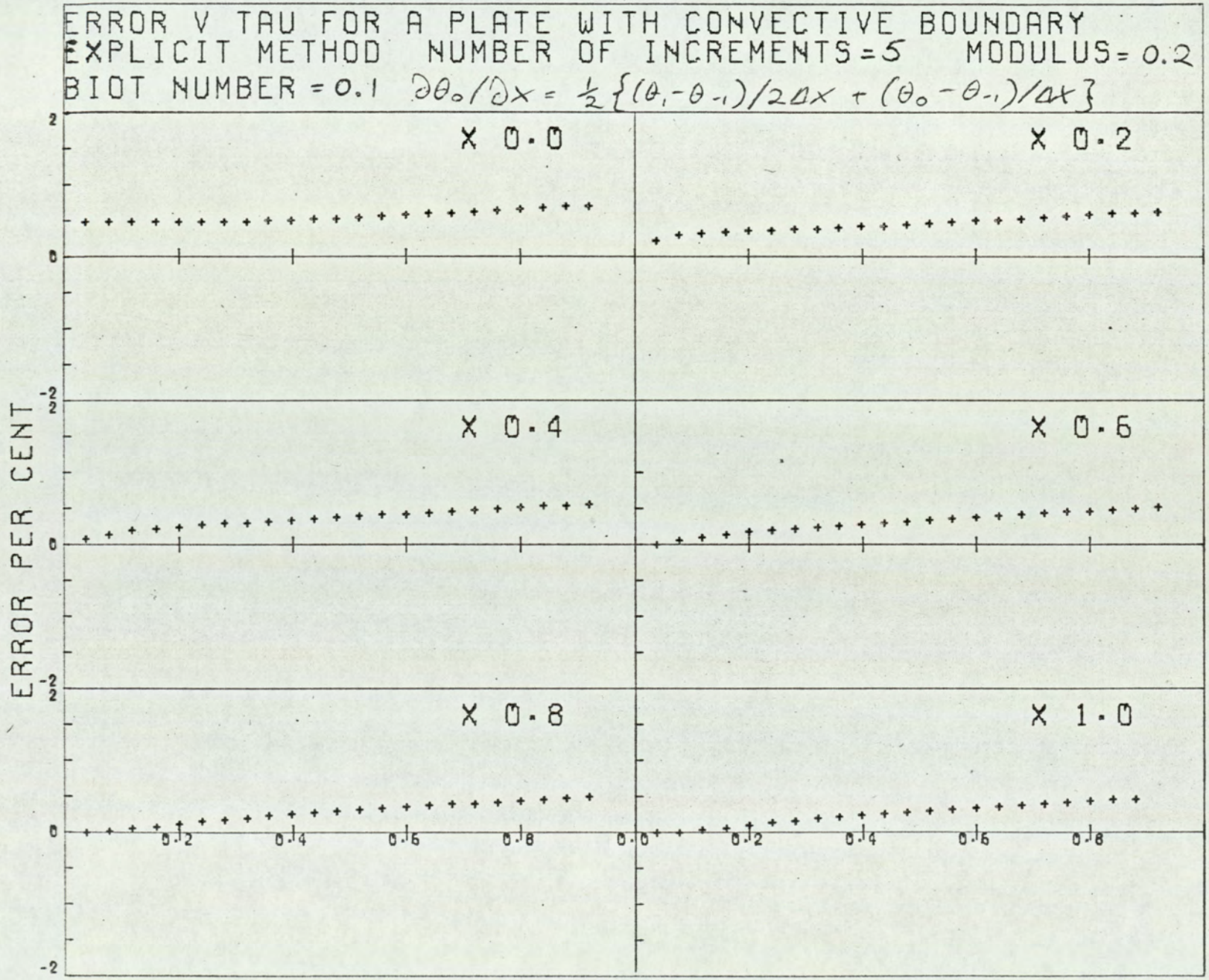

ERROR V TAU FOR A PLATE WITH CONVECTIVE BOUNDARY
EXPLICIT METHOD NUMBER OF INCREMENTS = 5 MODULUS = 0.2
BIOT NUMBER = 0.1 $\partial\theta_0/\partial x = \frac{1}{2}\{(\theta_1-\theta_{-1})/2\Delta x + (\theta_0-\theta_{-1})/\Delta x\}$
X 0·0
X 0·2
X 0·4
X 0·6
X 0·8
X 1·0
ERROR PER CENT
2
0
-2
0·0
0·2
0·4
0·6
0·8
TAU

FIG C.26

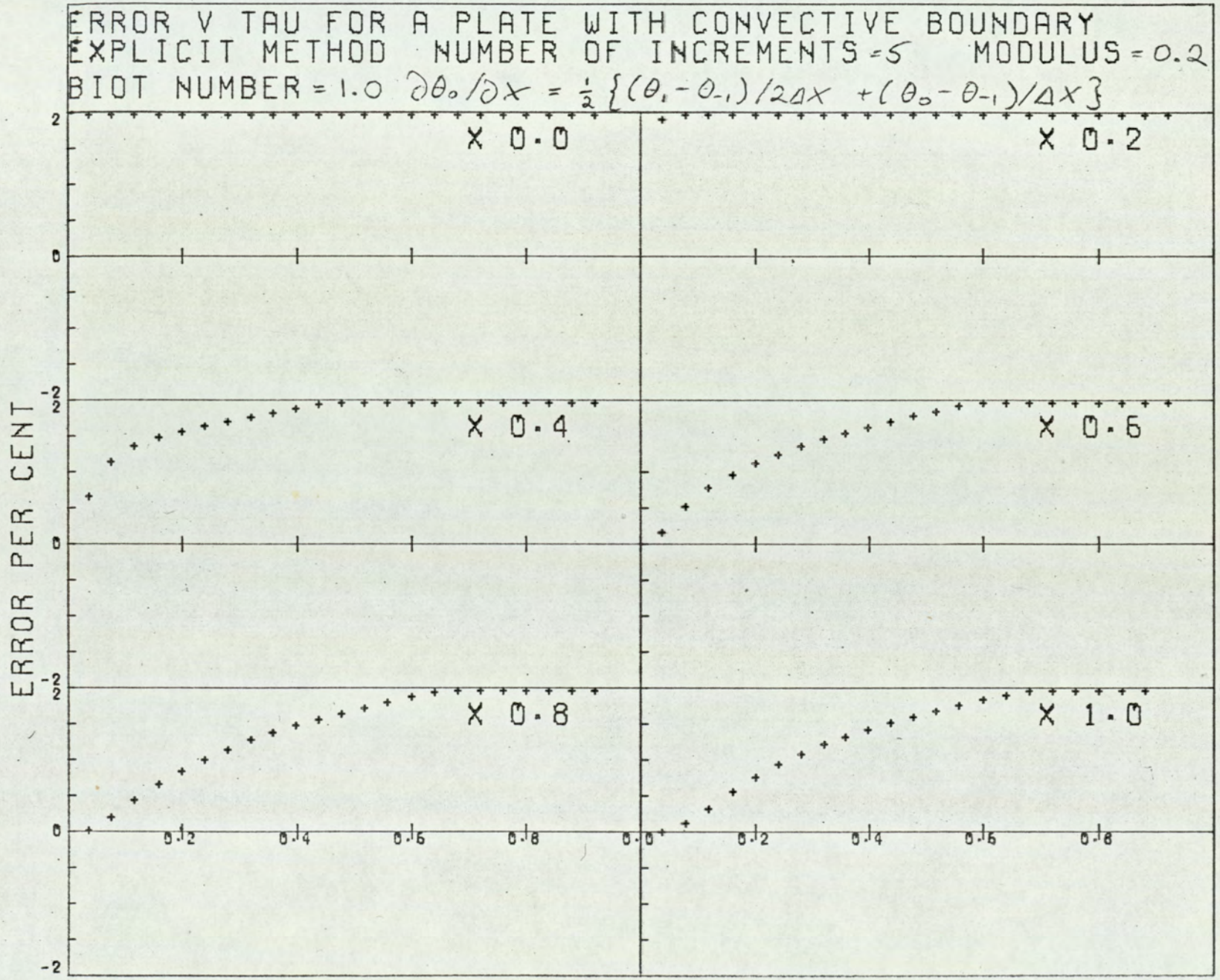

ERROR V TAU FOR A PLATE WITH CONVECTIVE BOUNDARY
EXPLICIT METHOD NUMBER OF INCREMENTS = 5 MODULUS = 0.2
BIOT NUMBER = 1.0 $\partial\theta_0/\partial x = \frac{1}{2}\{(\theta_1-\theta_{-1})/2\Delta x + (\theta_0-\theta_{-1})/\Delta x\}$
X 0.0
X 0.2
X 0.4
X 0.6
X 0.8
X 1.0
ERROR PER CENT
2
0
-2
0.2
0.4
0.6
0.8
0.0
TAU

FIG 6.27

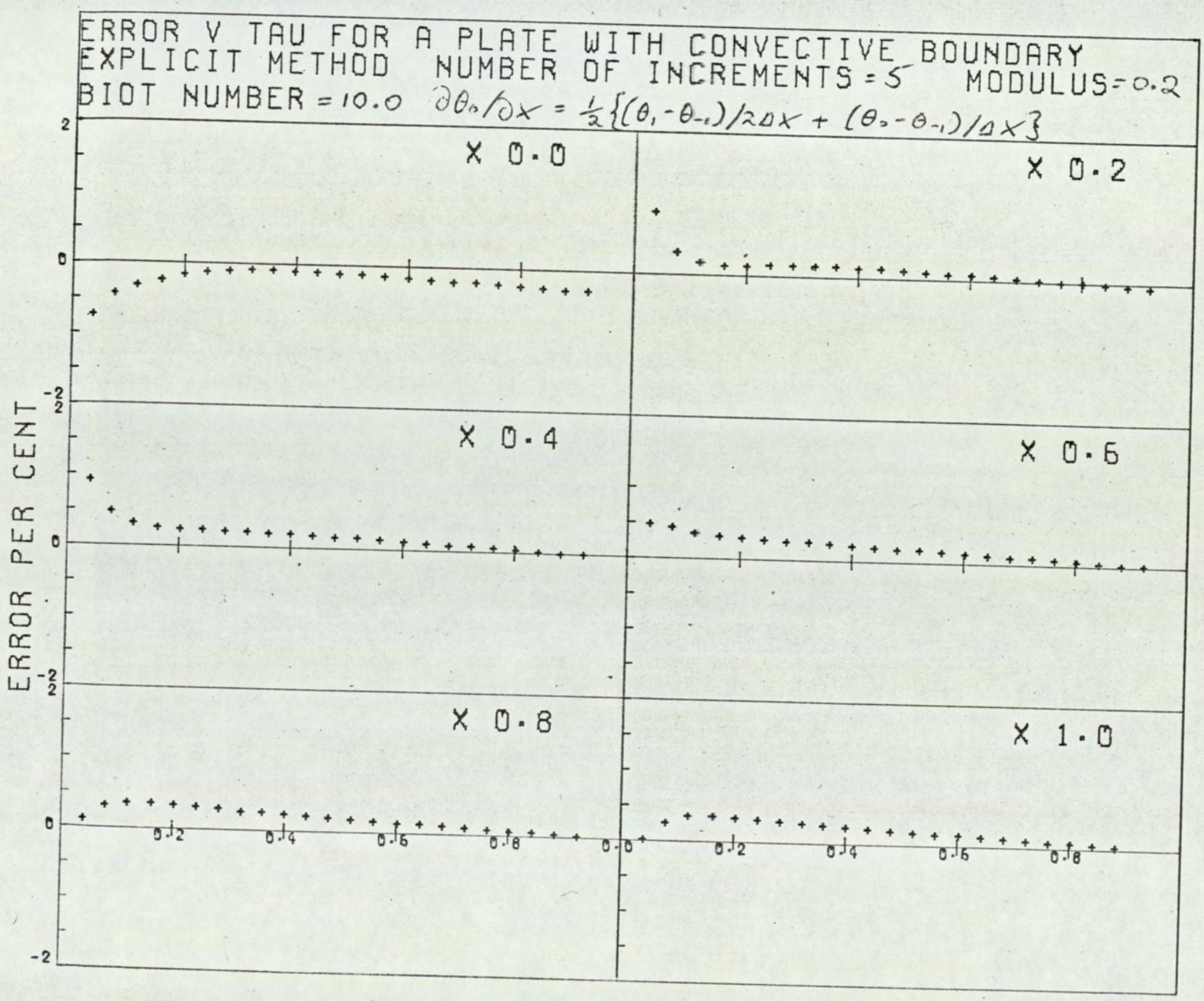

ERROR V TAU FOR A PLATE WITH CONVECTIVE BOUNDARY
EXPLICIT METHOD NUMBER OF INCREMENTS = 5 MODULUS = 0.2
BIOT NUMBER = 10.0 ∂θ0/∂x = ½{(θ1 - θ-1)/2Δx + (θ0 - θ-1)/Δx}
X 0.0
X 0.2
X 0.4
X 0.6
X 0.8
X 1.0
ERROR PER CENT
2
0
-2
0.2
0.4
0.6
0.8
1.0
TAU

FIG C 28

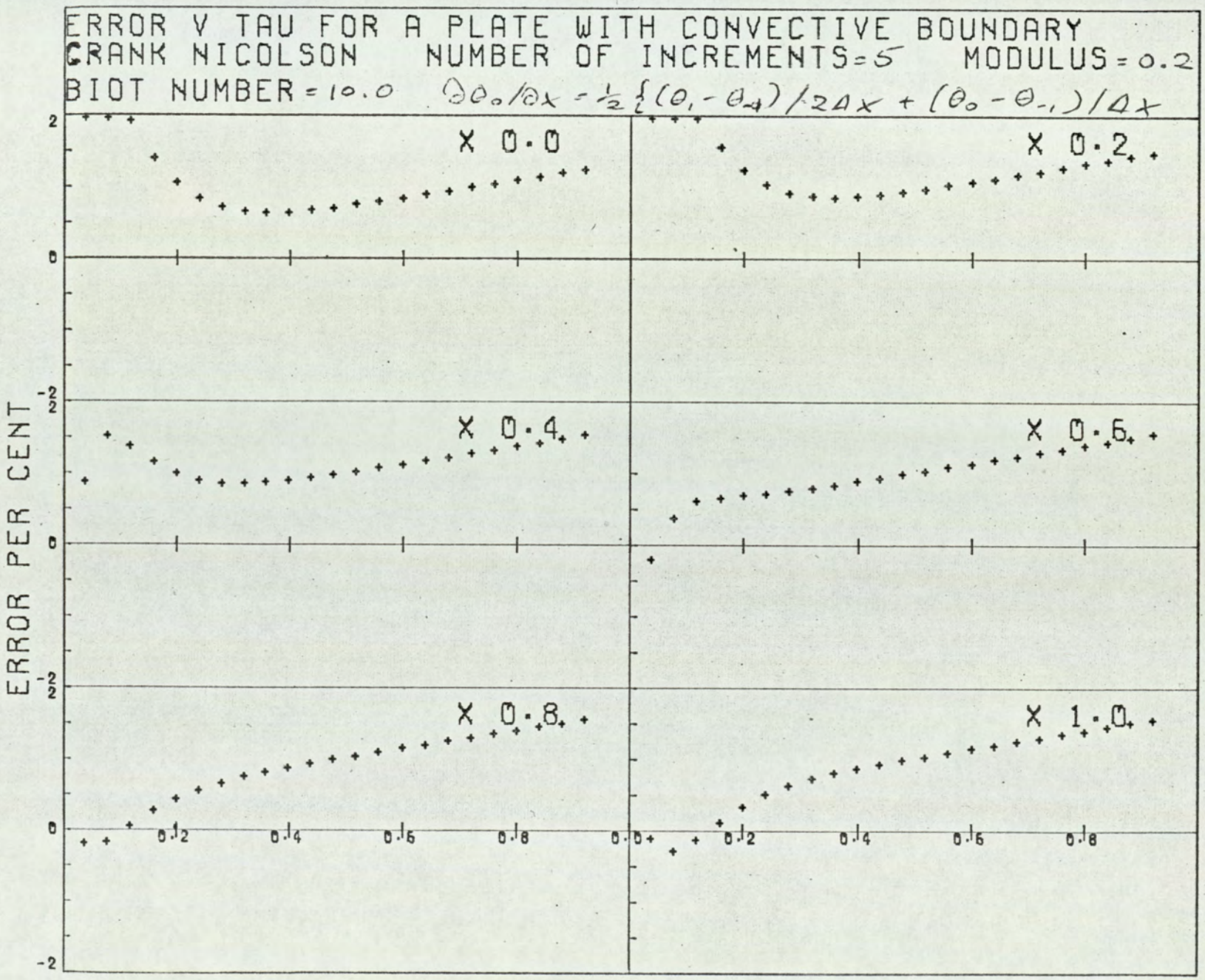

ERROR V TAU FOR A PLATE WITH CONVECTIVE BOUNDARY
CRANK NICOLSON NUMBER OF INCREMENTS=5 MODULUS=0.2
BIOT NUMBER=10.0 ∂θ0/∂x = ½{(θ1 - θ4)/2Δx + (θ0 - θ-1)/Δx
X 0.0
X 0.2
X 0.4
X 0.6
X 0.8
X 1.0
ERROR PER CENT
2
0
-2
0.2
0.4
0.6
0.8
TAU

FIG C.29

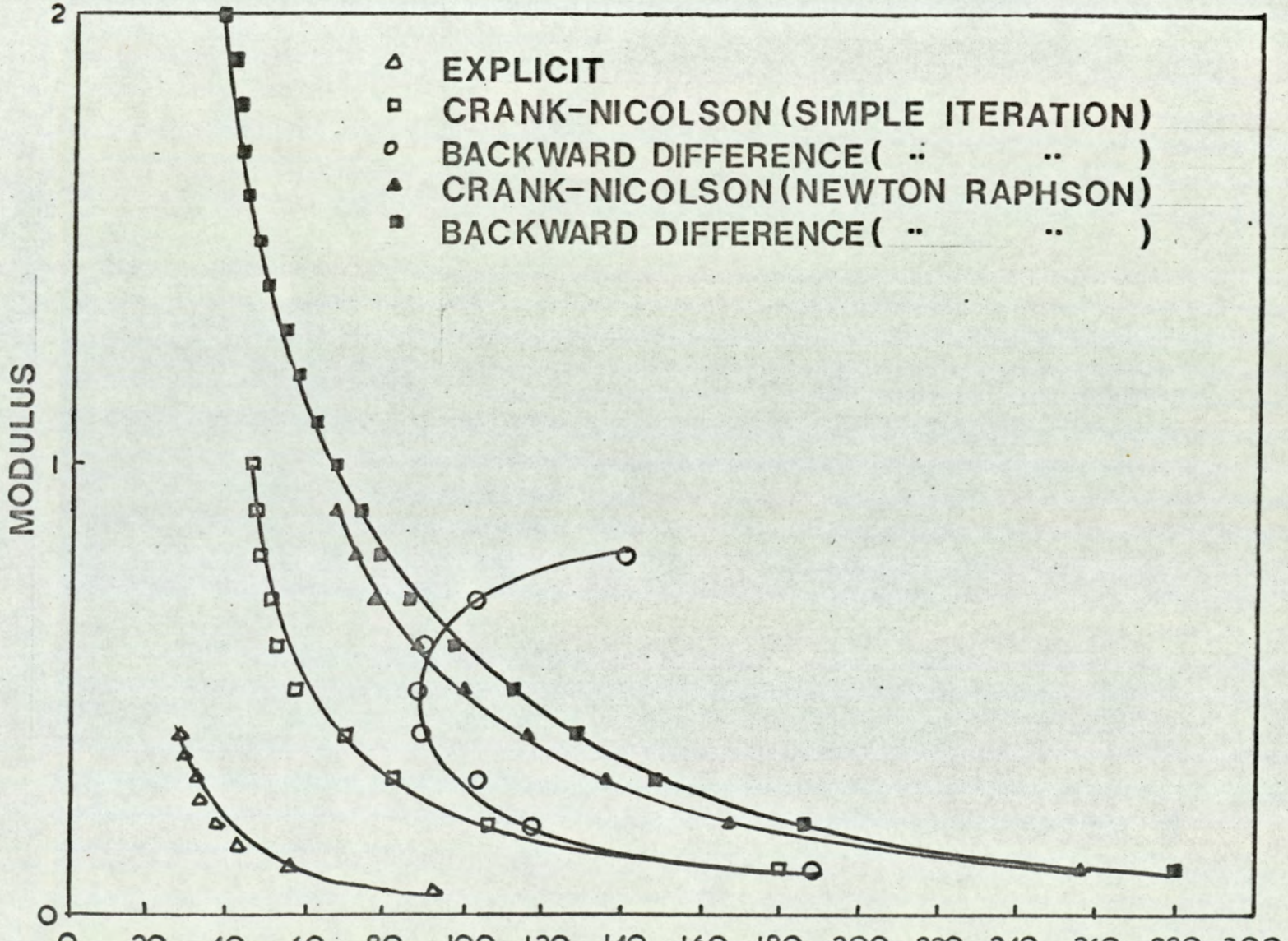

EXPLICIT
CRANK-NICOLSON (SIMPLE ITERATION)
BACKWARD DIFFERENCE(.. ..)
CRANK-NICOLSON (NEWTON RAPHSON)
BACKWARD DIFFERENCE(.. ..)
MODULUS
2
1
0
0 20 40 60 80 100 120 140 160 180 200 220 240 260 280 300
COMPUTATION TIME (SECS)

FIG C.30

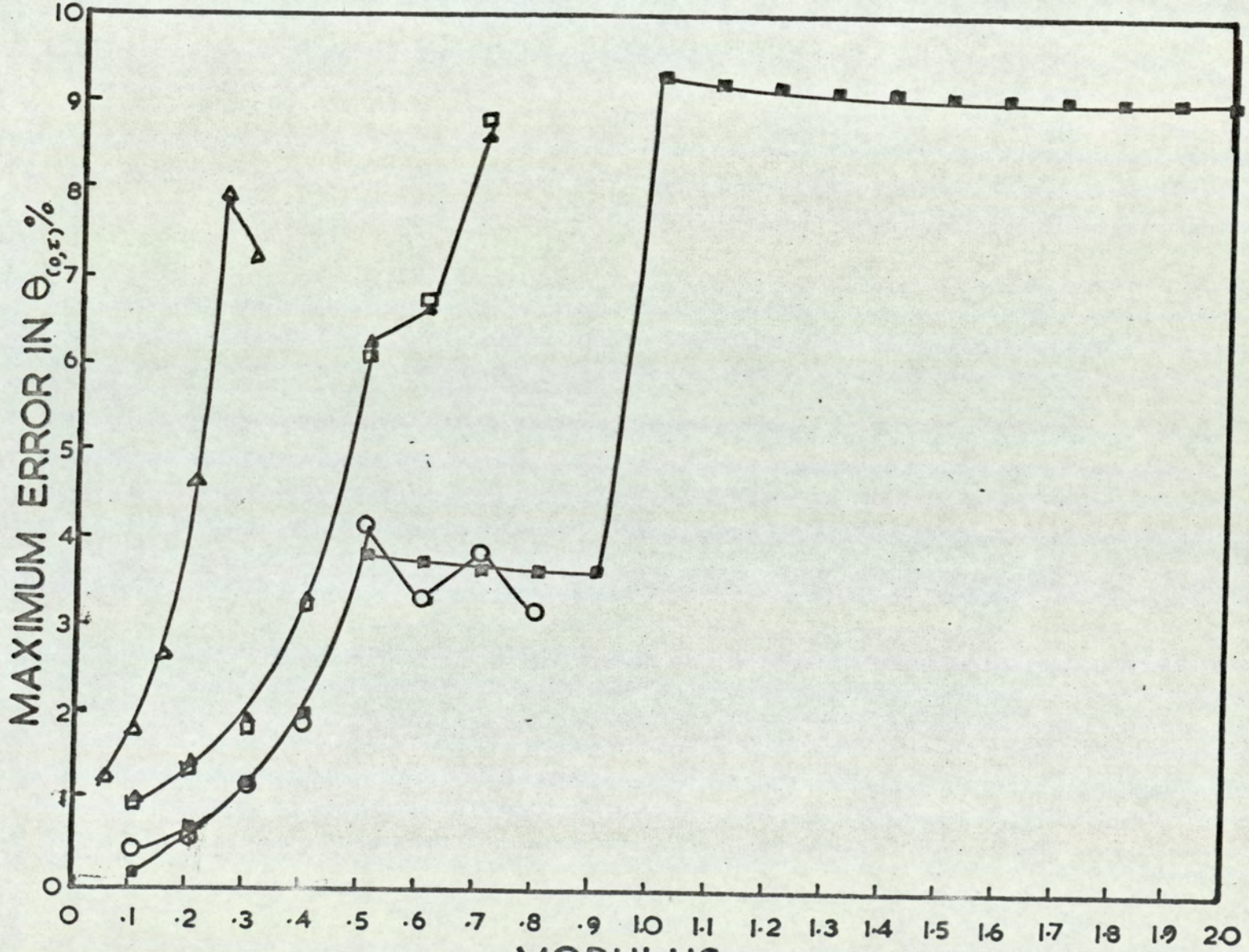


FIG C.31 RADIATING BOUNDARY

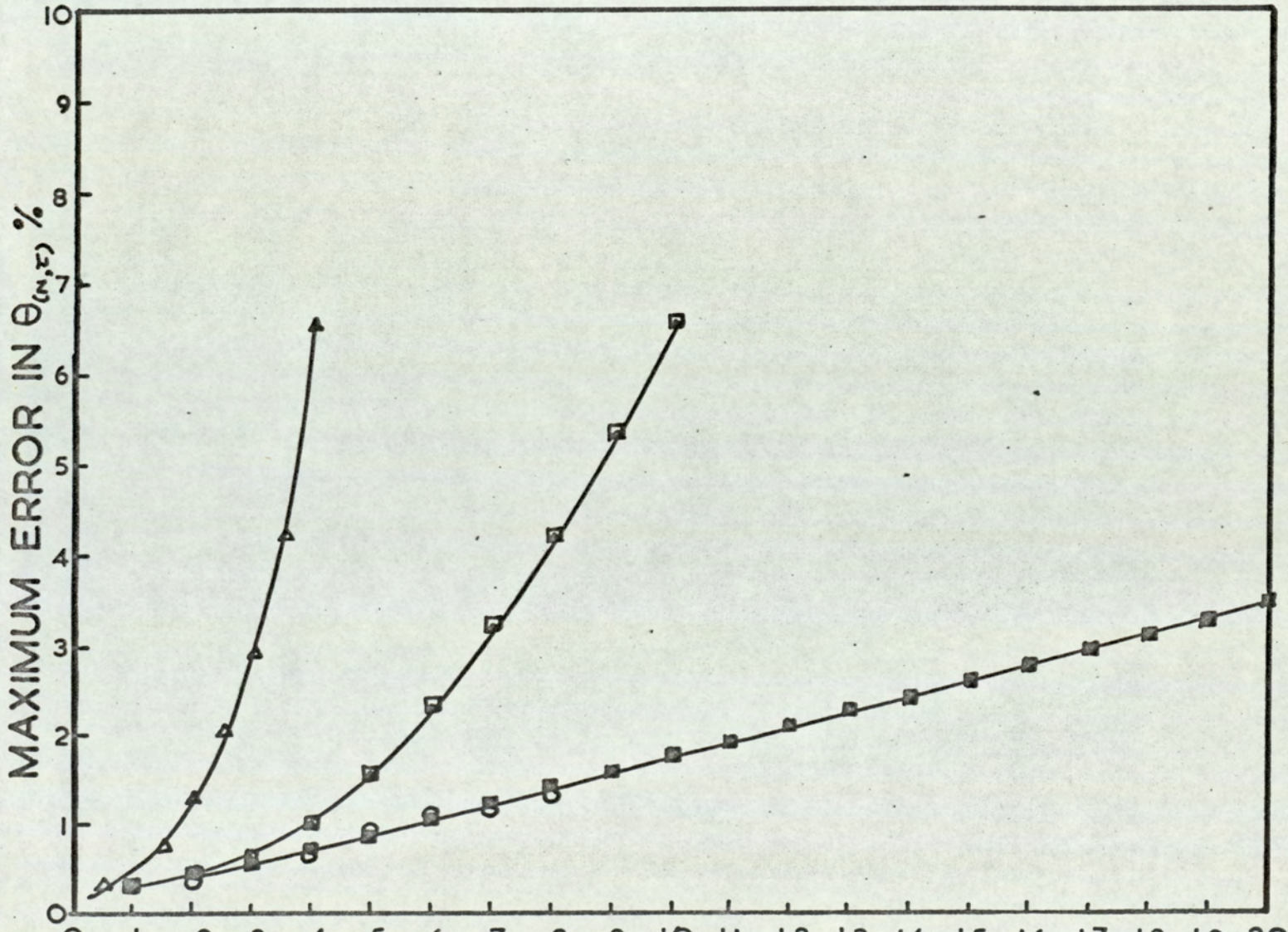


FIG C.32 INSULATED BOUNDARY

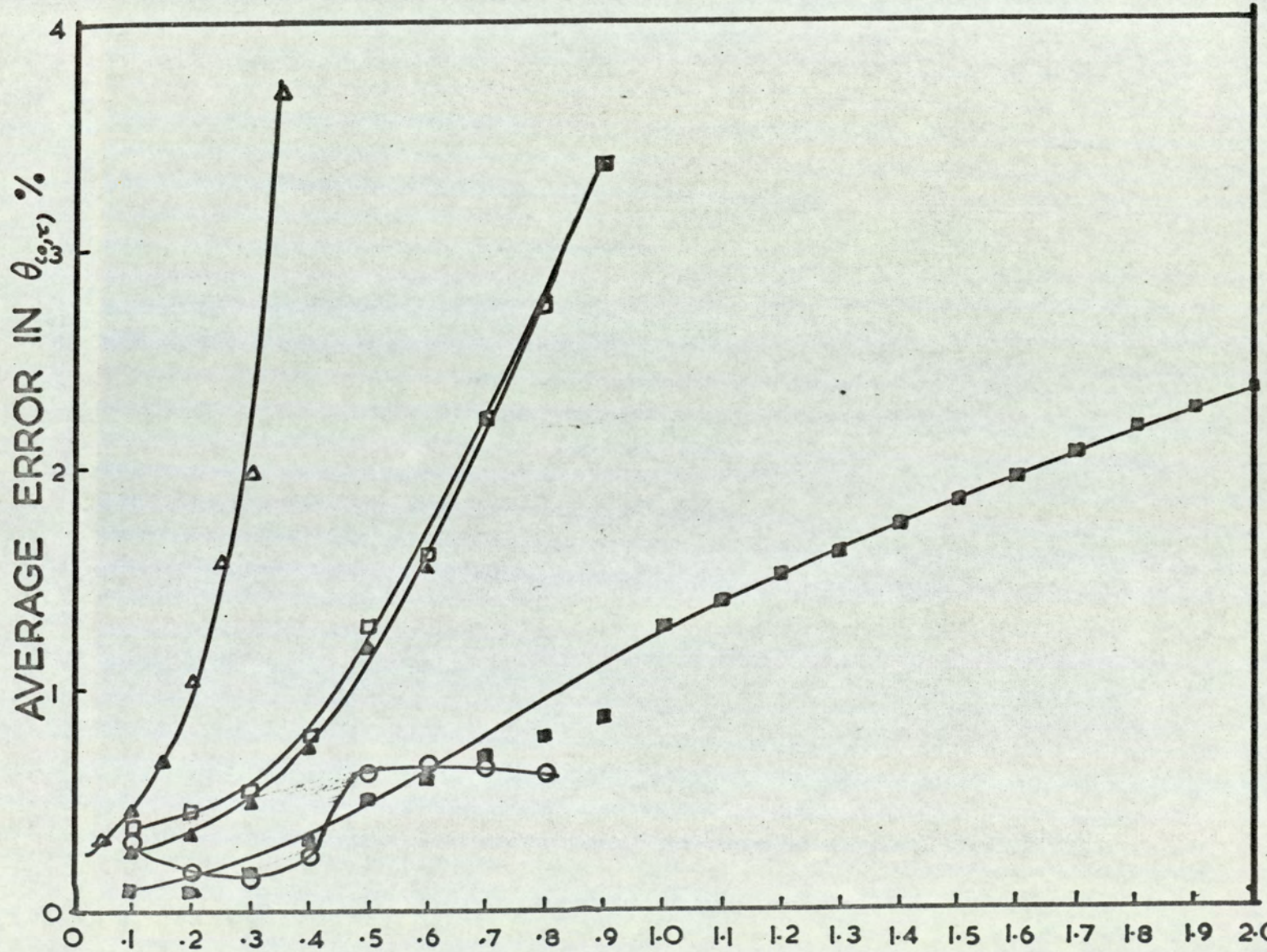


FIG C.33 RADIATING BOUNDARY

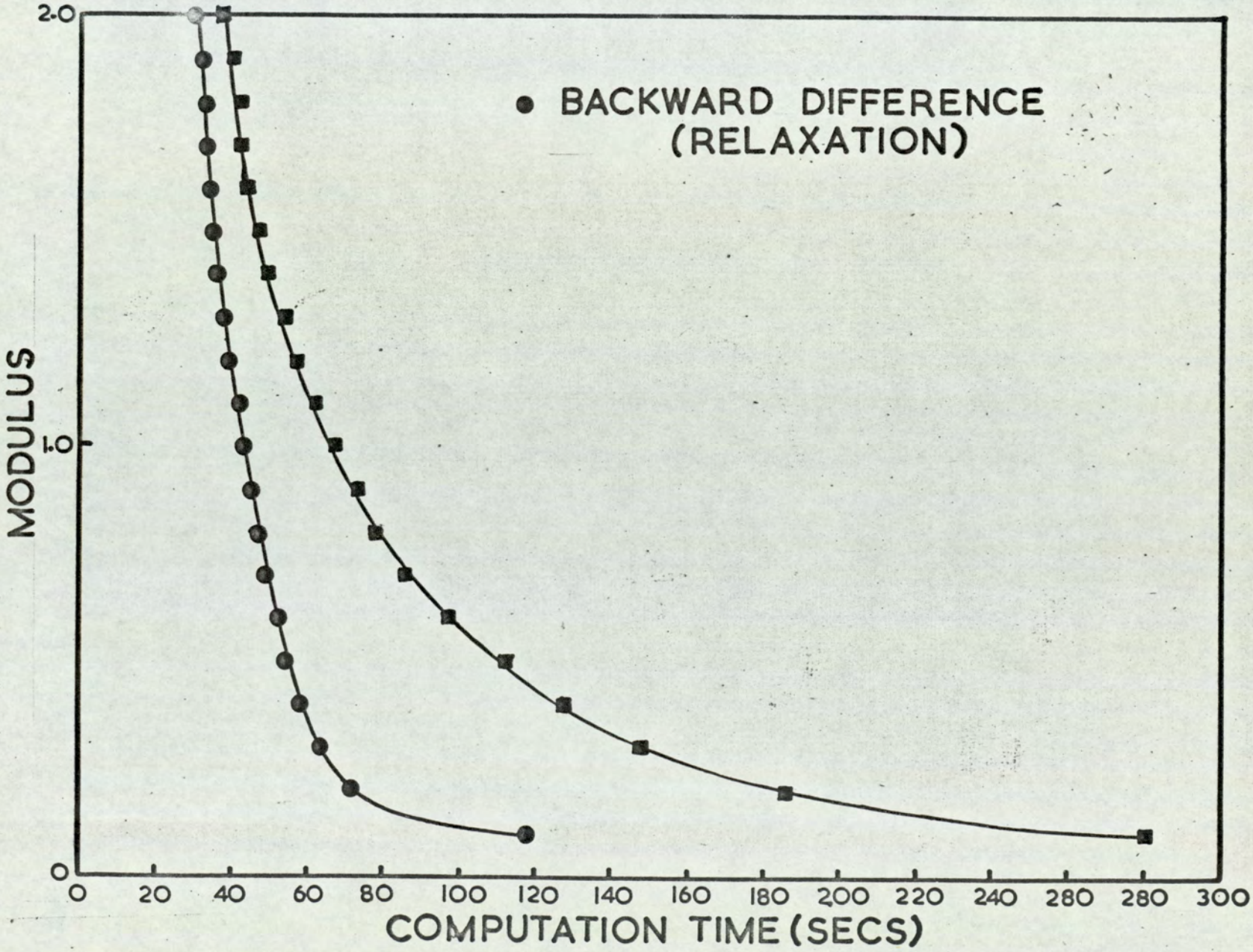


FIG C.34 SOLUTION TIMES FOR NEWTON RAPHSON AND RELAXATION METHODS $\omega = 0.7$

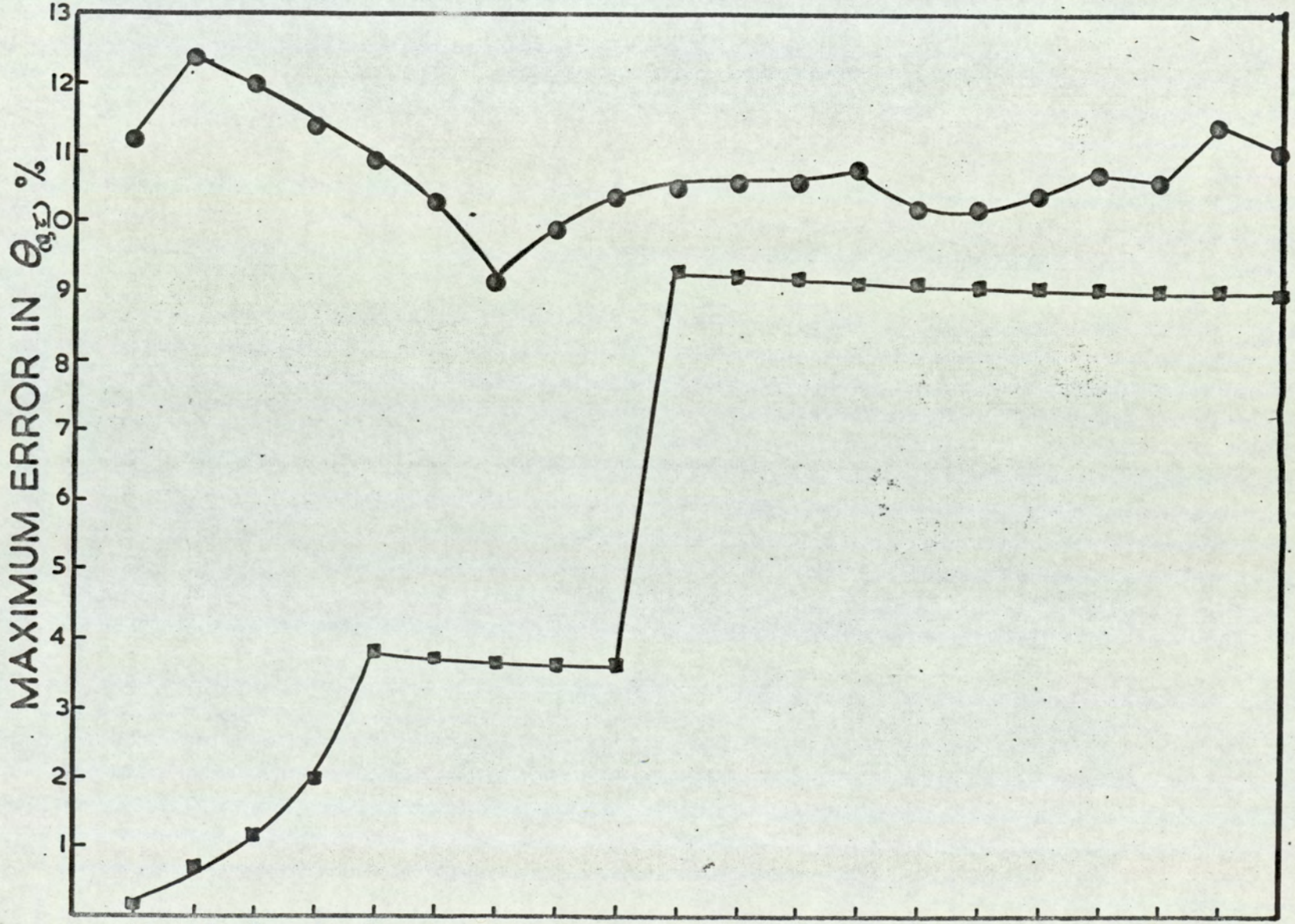


FIG C.35 ACCURACY OF NEWTON RAPHSON AND RELAXATION METHODS

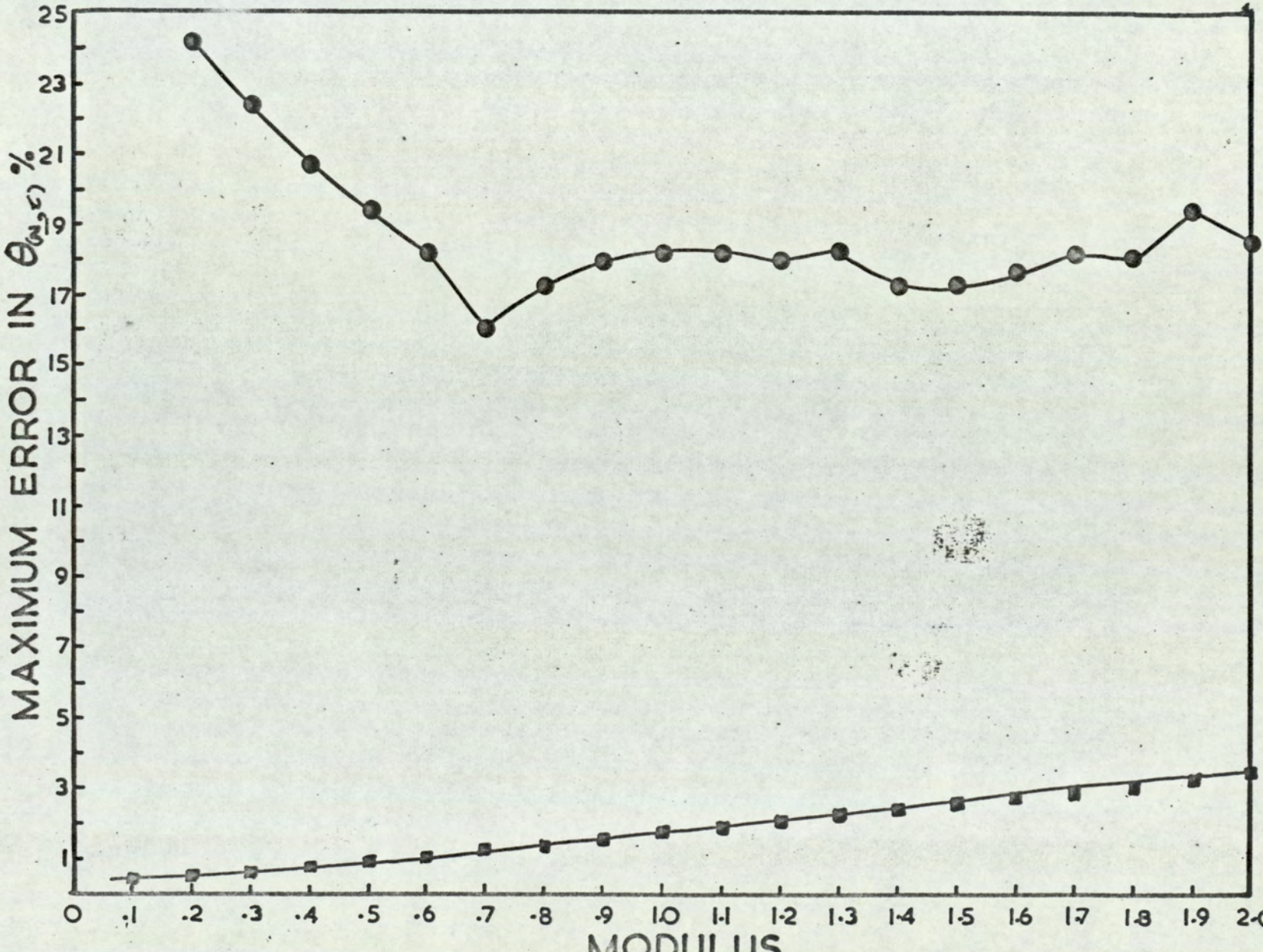


FIG C.36 ACCURACY OF NEWTON RAPHSON AND RELAXATION METHODS X=1 ω =0.7

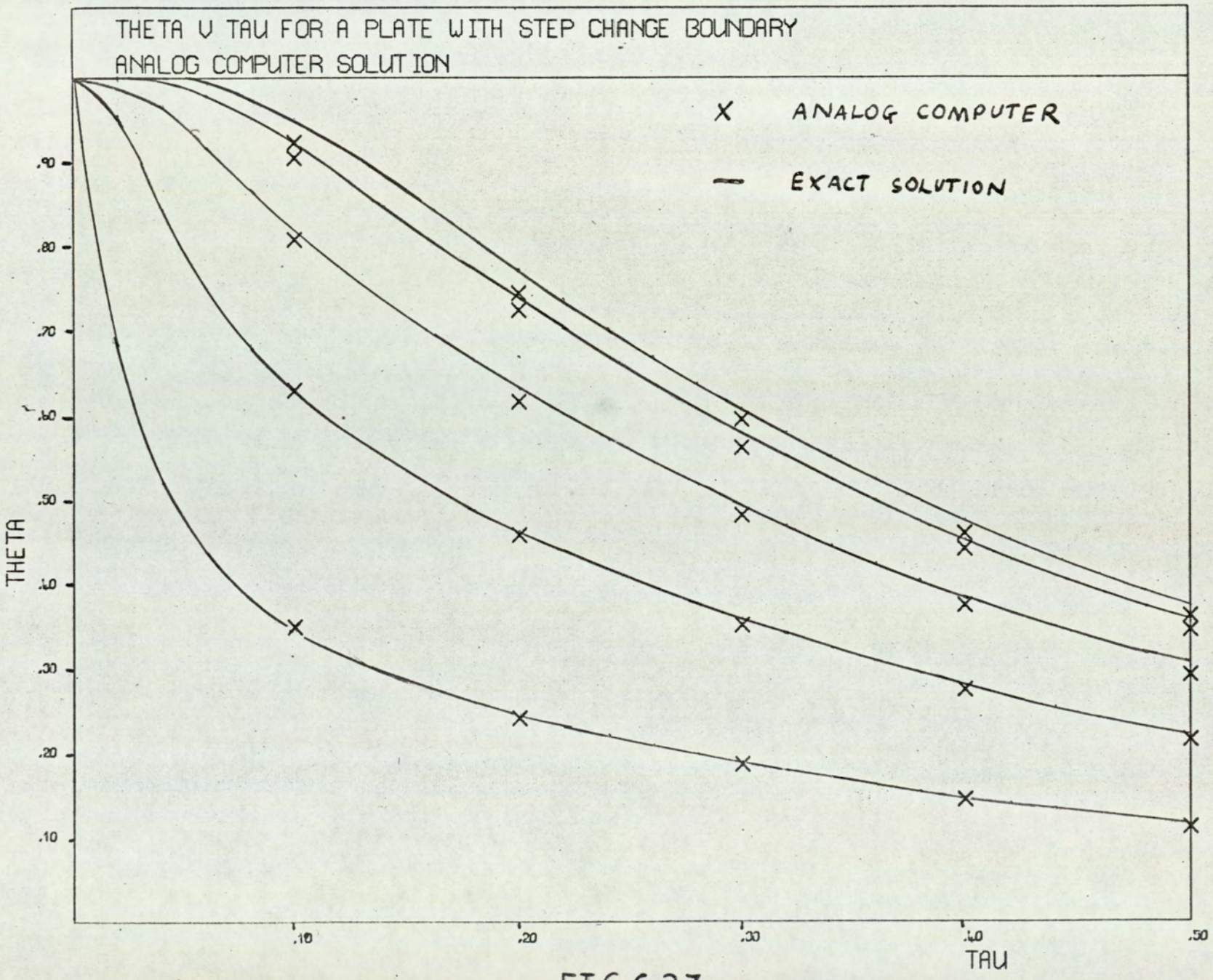

THETA V TAU FOR A PLATE WITH STEP CHANGE BOUNDARY
ANALOG COMPUTER SOLUTION
X ANALOG COMPUTER
— EXACT SOLUTION
THETA
.90
.80
.70
.60
.50
.40
.30
.20
.10
.10
.20
.30
.40
.50
TAU

FIG C 37

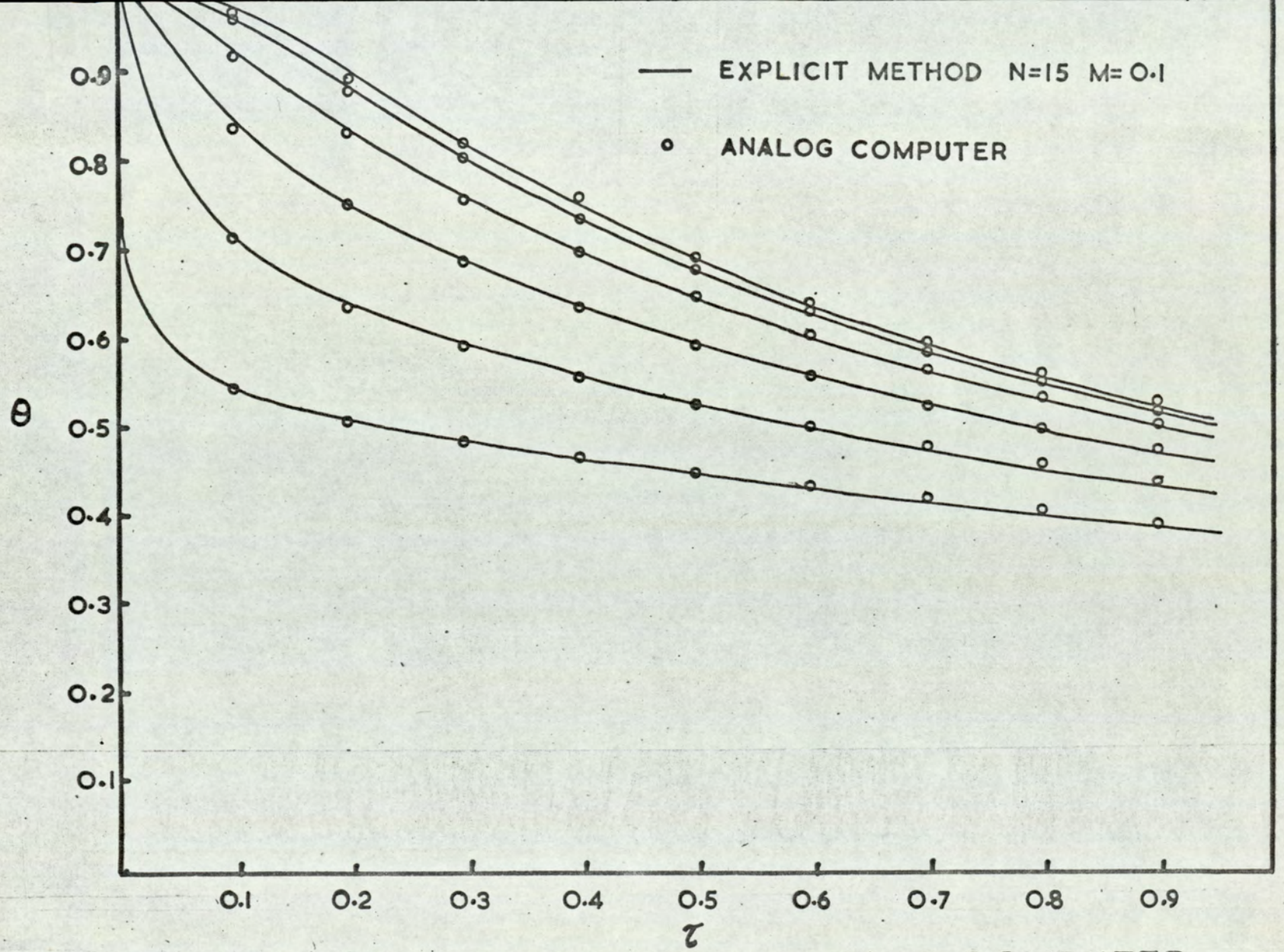


FIG C.38 ACCURACY OF THE ANALOGUE COMPUTER SOLUTION FOR A RADIATION BOUNDARY CONDITION R=10

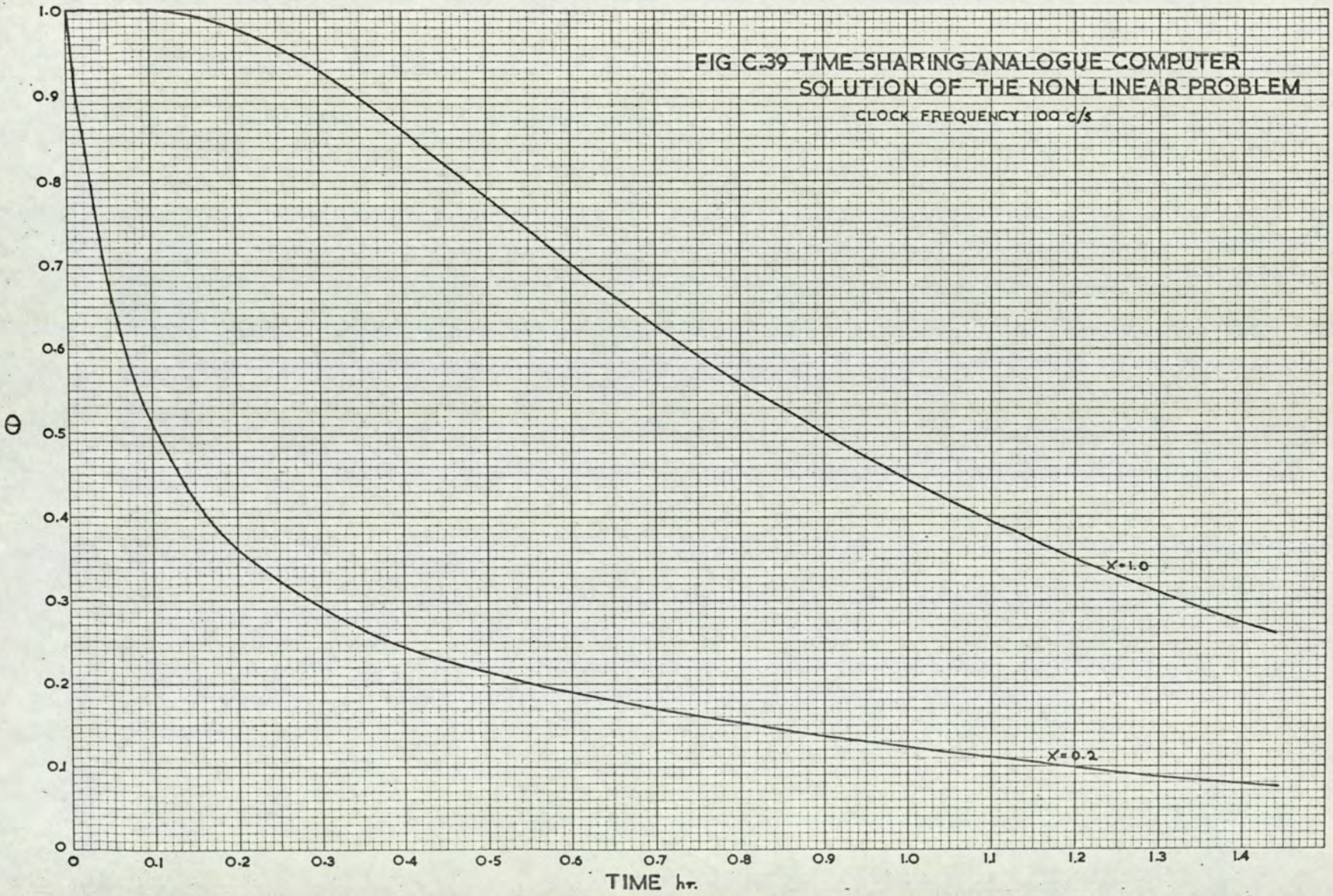


FIG C.39 TIME SHARING ANALOGUE COMPUTER
SOLUTION OF THE NON LINEAR PROBLEM
CLOCK FREQUENCY 100 c/s
θ
TIME hr.
X=1.0
X=0.2
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
1.1
1.2
1.3
1.4

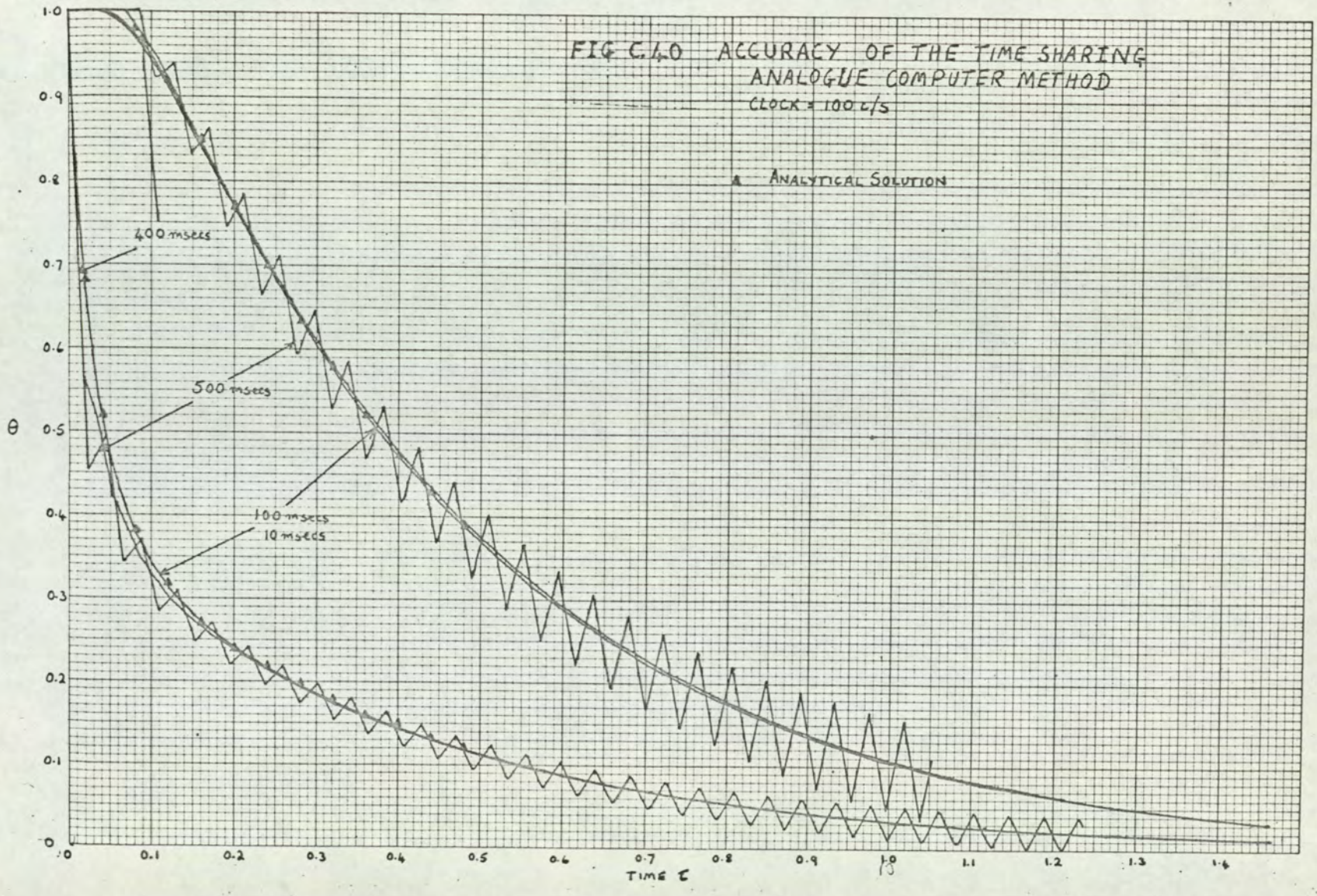


FIG C.4.0 ACCURACY OF THE TIME SHARING ANALOGUE COMPUTER METHOD
CLOCK = 100 c/s
Analytical Solution
400 msecs
500 msecs
100 msecs
10 msecs
θ
Time τ
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
1.1
1.2
1.3
1.4

REFERENCES

1. ALLADA, S. R., QUON, Donald. A stable explicit numerical solution of the conduction equation for multi-dimensional non-homogeneous media. Heat Transfer - Los Angeles. Chemical Engineering Progress. Symposium series 1966 62 151.

2. ANDERSON, J. T., BOTJE, J. M., KOFFEL, W. K. Comparison of the implicit and explicit methods for finite difference heat transfer calculations. J. Heat Transfer. 1961 83 516.

3. ANDERSON, J. T., BOTJE, J. M., KOFFEL, W.K. Digital computer solution of complex transient heat transfer problems. Engineering Experimental Station Bulletin, West Virginia University, 1961.

4. Back, L. H. Numerical approximations of the convective boundary condition. J. Heat Transfer. 1962 84 89.

5. CAMPBELL, D. J., VOLLENWEIDER, D. B. Unusual techniques applied in heat transfer programs. 1959 Proceedings of the Eastern Joint Computer Conference.

6. CHAPMAN, A. J. Heat Transfer. The Macmillan Co. 1960 (New York).

7. CRANK, J., NICOLSON, P. A practical method for the numerical evaluation of solutions of partial differential equations of the heat conduction type. Proc. Camb. Phil. Soc. 1947 41 50.

8. DOUGLAS, J. JR. A survey of numerical methods for parabolic differential equations. Advances in Computers 1962 2 1.

9. DOUGLAS, J. JR. On the numerical integration of $\partial u/\partial t = \partial^2 u/\partial x^2 + \partial^2 u/\partial y^2$ by implicit methods. J.S.I.A.M. 1955 3 (1) 42.

10. DOUGLAS, J. JR., RACHFORD, H.H. On the numerical solution of heat conduction problems in two and three space variables. Trans. Amer. Math. Soc. 1956 82 421.

11. DUFORT, E. C., FRANKEL, S. P. Stability conditions in the numerical treatment of parabolic differential equations. Math. Tables. Aids Computation. 1953, 7 135.

12. DUSINBERRE, G. M. Numerical methods for transient heat flow. Trans. Amer. Soc. Mech. Engrs. 1945 67 703.

13. DUSINBERRE, G. M. A note on the implicit method for finite difference heat transfer calculations. J. Heat Transfer 1961 83 94.

14. Electronic Associates Ltd. Parallel hybrid computer applications. No. 3. The solution of partial differential equations by time-sharing using multiplexing techniques.

15. ELROD, H. G. Jr. New finite difference techniques for the solution of the heat conduction equation especially near surfaces with convective heat transfer. Trans. Amer. Soc. Mech. Engrs. 1957 79 1519.

16. FAIRALL, R. S., WELLS, R. A., BELCHER, R. L. Unsteady state heat transfer in solids with radiation at one boundary. J. Heat Transfer. 1962 82 266.

17. FOX, D. W., SHAW, H. Jr., JELLINEK, J. Numerical approximations in heat transfer problems and usage of IBM 7090 computer for solutions. John Hopkins Univ., Silver Spring Md., Appl. Phys. Lab., Contract NOrd. 7386, 17th May 1962 51pp.

18. FRANKEL, S. P. Convergence rates of iterative treatments of partial differential equations. Math. Tables Aids Computation 1950 4 65.

19. GAY, B. Comparison of methods for solution of the heat conduction equation with a radiation boundary condition. Int. J. Heat Mass Transfer. 1965 8 507.

20. GOODMAN, T. R. Application of integral methods to transient non-linear heat transfer. Advances in Heat Transfer 1964 1 51.

21. GREENWOOD, J. A. Implicit numerical methods for the heat conduction equation. Brit. J. Appl. Phys. 1962 13 (11), 571.

22. HARTREE, S., WOMERSLEY, J. R. A method for the numerical or mechanical solution of certain types of partial differential equations. Proc. Royal Soc. London, Series A, 161 (1937) 353.

23. JENSON, V. G., JEFFREYS, G. V., Mathematical methods in chemical engineering. A.P.

24. KARDAS, A. Errors in a finite difference solution of the heat flow equation. J. Heat Transfer. 1964 86 561.

25. KEAST, P., MITCHELL, A. R. On the instability of the Crank Nicolson formula under derivative boundary conditions. The Computer J. 1966 9 110.

26. LAPIDUS, L. Digital computation for chemical engineers. Mc Graw-Hill.

27. LIEBMANN, G. The solution of transient heat flow and heat transfer problems by relaxation. Brit. J. Appl. Phy. 1955 6 129.

28. LYNN, L. L. MEYER, J. E. A numerical comparison of the implicit and explicit techniques for the convective boundary condition. J. Heat Transfer. 1963 85 280.

29. MODERN COMPUTING METHODS. Notes on Applied Science No. 16. HMSO (London, England). 1961. 93.

30. O'BRIEN, G. G., HYMAN, M. A., KAPLAN, S. A study of the numerical solution of partial differential equations. J. Maths. Phys. 1950 29 223.

31. PEACEMAN, D. W., RACHFORD, H. H. Jr. The numerical solution of parabolic and elliptic differential equations. J. Soc. Ind. Appl. Math. 1955 3 28.

32. PEARSON, C. E. Impulsive and conditions for diffusion equation. Math. Tables Aids. Computation. 1965 19 570.

33. RICHARDSON, L. F. The approximate arithmetical solution by finite differences of physical problems involving differential equations. Phil. Trans. R.S. London, Series A 1910 210 307.

34. RICHTMEYER, R. D. Difference methods for initial value problems. Interscience Publishes Inc. New York 1957.

35. REYNOLDS, W. C., DOLTON, T. A. Technical Report No.36. Prepared under contract No 225(23) (NR-090-342). For Office of Naval Research.

36. ROBERTS, A. F. Heating of cylinders by radiation: an approximate formula for the temperature distribution. ASME Paper 64-HT-39.

37. SCHNEIDER, P. J. Temperature response charts. 145-146. Wiley, New York (1963).

38. SCHNEIDER, P. J. Radiation cooling of finite heat conducting solids. J. Aerospace Sci. 1960 27 546.

39. SIDDALL, R. G. Simple approximate solutions of the equation of heat conduction. Fuel Soc. J. (Sheffield Univ.) 1962 13 19.

40. VARGA, R. S. Matrix Iterative analysis. Prentice-Hall Inc. New Jersey 1963.

41. WAGNER, R. J. Heat 1 - a one dimensional time dependent or steady state heat conduction code for the IBM 650. Phillips Petrol Co. Atom. Ener. Dur., Idaho Falls, Idaho. IDO-16867, Contract AT(10-1) -205, Apr. 1963 76pp. OTS 1.75.

42. ZERKE, R. D., SUNDERLAND, J. E. The transient temperature distribution in a slab subject to thermal radiation. J. Heat Transfer. 1965 87 117.

LIST OF FIGURES

C.14 " explicit method N = 5 M = 0.5 Bi = 1 equ. (1.33)

C.15 " " " N = 5 M = 0.2 Bi = 10 " "

C.16 " " " N = 10 M = 0.2 Bi = 10 " "

C.17 " " " N = 10 M = 0.4 Bi = 10 " "

C.18 " " " N = 15 M = 0.2 Bi = 10 " "

C.19 " Crank-Nicolson N = 5 M = 0.2 Bi = 1 " "

C.20 " " " N = 5 M = 1.4 Bi = 1 " "

C.21 " " " N = 5 M = 0.2 Bi = 10 " "

C.22 " " " N = 10 M = 1.4 Bi = 10 " "

C.23 " " " N = 10 M = 0.2 Bi = 10 " "

C.24 " Explicit method N = 5 M = 0.1 Bi = 0.1 " (1.32)

C.25 " " " N = 5 M = 0.2 Bi = 10 " "

C.26 " " " N = 5 M = 0.2 Bi = 0.1 equ.(2.1)

C.27 " " " N = 5 M = 0.2 Bi = 1.0 " "

C.28 " " " N = 5 M = 0.2 Bi = 10 " "

C.29 " Crank-Nicolson N = 5 M = 0.2 Bi = 10 " "

C.30 Solution for a plate with radiation and adiabatic boundary conditions (Section 2.L) Modulus compared with computation time.

C.31 " Maximum error at the radiating boundary as a function of modulus.

C.32 " Maximum error at the insulated boundary as a function of modulus.

C.33 " Average error at the radiating boundary as a function of modulus.

C.34 Solution times to achieve a convergence of 0.005 using relaxation for a plate with radiation and adiabatic boundary conditions.

C.35 Accuracy of the Newton-Raphson and relaxation methods for a plate with radiation and adiabatic boundary conditions. Radiating boundary.

C.36 " Insulated boundary.

C.37 Analogue computer solution for a plate with step change and adiabatic boundary conditions compared with the analytical solution.

C.38 Analogue computer solution for a plate with radiation and adiabatic boundary conditions compared with the explicit "exact" solution R = 10.

C.39 Analogue computer solution for the non-linear problem using the time-sharing method.

C.40 Accuracy of the time-sharing method.

SYMBOLS

A, a	
B, b	
C, c	
D, d	Constants
V	
β	
δ	
Bi	Biot number = $h.L/k$
C	Specific heat
C'	Scaled value of C
E	Error
$\hat{E}$	Maximum error %
e	Emissivity
h	Heat transfer coefficient
K	Parameter in summary equation
k, k_x	Thermal conductivity of material in x direction
k_y	Thermal conductivity of material in y direction
k_z	Thermal conductivity of material in z direction
k'	Scaled value of k
k	Subscript
L, L_x	Thickness of material in x direction
M	Modulus = $\Delta\tau / \Delta x^2$
m	Iteration number

N	Number of space increments
n	Subscript
P	% number of points with error 2%
R	Radiation parameter = $L\sigma\epsilon T_i^3 / k$
S	Special matrix
s	1/M
T	Temperature
T_i	Initial temperature
T_o	Radiating
T_s	Sink temperature
T_∞	Temperature to which material tends when
T_{xp}	Value of T_o when the temperature at the boundary X = 1 starts to change.
t	Time
Δt	Time step
X	Dimensionless distance = x / L_x
x	Distance into material from heat transfer boundary in first dimension
x_p	Penetration distance; the value of x at which the temperature just starts to alter
Y	y / L_y
y	Distance into material from heat transfer boundary in second dimension
Z	z / L_z
z	Distance into material from heat transfer boundary in third dimension
α	Thermal diffusivity = $k / \rho C$

σ	Stefan - Boltzmann constant
θ	Dimensionless temperature = $(T-T_\infty)/(T_i-T_\infty)$
τ	Dimensionless time = $\alpha t/L^2$
τ_{xp}	Value of τ when $\theta_o = \theta_{xp}$
$\Delta\tau$	Step length in τ direction