MITNE - 203

IMPROVED MULTIDIMENSIONAL NUMERICAL METHODS FOR THE STEADY STATE AND TRANSIENT THERMAL-HYDRAULIC ANALYSIS OF FUEL PIN BUNDLES AND NUCLEAR REACTOR CORES

by

Robert E. Masterson and Lothar Wolf

June, 1977

DEPARTMENT OF NUCLEAR ENGINEERING
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
Cambridge, Massachusetts 02139

MITNE-203

IMPROVED MULTIDIMENSIONAL NUMERICAL METHODS FOR THE STEADY STATE AND TRANSIENT THERMAL-HYDRAULIC ANALYSIS OF FUEL PIN BUNDLES AND NUCLEAR REACTOR CODES

by

Robert E. Masterson and Lothar Wolf

June, 1977

Department of Nuclear Engineering
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

ABSTRACT

The purpose of this report is to describe the development of two new and extremely efficient numerical methods for the steady state and transient, two-phase, lumped parameter thermal hydraulic analysis of the fluid flow distributions in fuel pin bundles and nuclear reactor cores as the result of ongoing research in the Department of Nuclear Engineering at M.I.T.

These methods use the same physical model as the COBRA-IIIC code, but are based on the alternative numerical concept of generating a system of semi-implicit difference equations for the pressure field using a spatial differencing scheme which is different from the schemes previously used by sub-channel analysis codes. The flow and enthalpy distributions in the lattice are found by marching downstream several times in succession between adjacent computational planes and by combining the computed pressure fields from these planes together into a composite pressure field, which is then used as the driving force for the crossflow distribution in a reformulated form of the transverse momentum equation. Both methods are extremely efficient from a computational point of view and are compatible with a variety of iterative solution techniques because the coefficient matrices governing the computation of the pressure fields can be shown to be Stieltjes matrices.

These numerical methods have been integrated into the computational framework of the COBRA-IIIC code and a new computer program has been written called COBRA-IIIP/MIT (P for a pressure solution). The code is considerably faster and more powerful than many other thermal-hydraulic analysis codes and has the capability to solve extremely large and complex problems with great speed. The predictions of the code have been compared to those of COBRA-IIIC, and for most problems, it has been found that the results of the codes agree with one another to within a few tenths of one percent. However, it has been discovered that the axial iteration scheme used by the COBRA-IIIC code apparently introduces severe oscillations into the crossflow distribution that cannot be explained physically, whereas the numerical solution scheme used by the COBRA-IIIP/MIT code does not appear to suffer from this drawback and converges very rapidly to an asymptotic crossflow distribution without oscillations as additional iterations are performed.

It is strongly recommended that the COBRA-IIIP/MIT code should be used in place of COBRA-IIIC for the analysis of operational reactor conditions, because of its greater computational efficiency, and that experiments should be performed to determine the range of applicability of these codes when they are applied to model severe loss-of-flow or overpower transients. The COBRA-IIIP/MIT code and a user's manual describing its computational capabilities will be furnished to the Argonne National Code Center. The code will be made available for public distribution in 1978.

ACKNOWLEDGEMENTS

The work described in this report was performed primarily by the principal author, Robert E. Masterson, as a graduate student in the Department of Nuclear Engineering during the fall of 1976 and the spring and summer of 1977 in partial fulfillment of the requirements for the degree of Doctor of Science at M.I.T.

The advice and suggestions of Mr. Pablo Moreno, Professor Neil Todreas, and many of the other students and faculty at M.I.T. are gratefully acknowledged. Gratitude is also expressed to Ms. Rachel Morton for her advice on computational problems, to Ms. Virginia O'Keefe for her administrative support, and to the principal author's wife, Carol Craig Masterson, who very capably handled the typing of this manuscript.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER I

1.1 Research Objective

In recent years many attempts have been made to develop numerical methods for solving the set of fluid conservation equations which characterize the steady state and transient thermal-hydraulic performance of fuel pin bundles and nuclear reactor cores during both single phase and two phase flow conditions. These equations are usually developed from a control volume approach before they are cast into a form more suitable for digital computer analysis. The purpose of this thesis is to present two numerical methods for solving these equations which are considerably faster and more efficient than previous methods used to analyze operational reactor conditions. Another goal of this thesis is to develop a computer code based on these methods which can be used to solve a variety of complex and important problems that arise in the thermal-hydraulic analysis of light water reactor cores.

1.2 Introduction

An excellent review of the state of the art of reactor thermal-hydraulic analysis codes has been given by Weisman and Bowring [1]. Many of these codes have attempted to simplify the solution of the conservation equations by dividing the lattice into a number of computational cells having the same size and shape [2, 3, 4], and by using an initial value approach

to alleviate the numerical difficulties associated with solving the conservation equations as a true boundary value problem in which arbitrary boundary conditions are imposed on the pressure and flow fields at the inlet and outlet of the core. These simplifications have enabled subchannel analysis codes to handle fuel pin bundles having as many as 37 rods [5], although some of the most widely used and acknowledged codes, such as HAMBO [6] and COBRA-IIIC [7] have generally been limited to problems having much smaller numbers of computational cells.

The COBRA-IIIC code [7] is a modified version of the COBRA-II code [8] in which a backward differencing scheme is used for the momentum equations to propagate information to upstream as well as downstream locations. The COBRA-IIIC code has the capability to compute the flow and enthalpy fields in fuel pin bundles during steady state and transient conditions, and has been frequently applied to problems containing flow blockages with both single phase and 2-phase flow. The primary drawback of the code is that it can only be used to handle problems having a moderate number of (i.e. 30-40) computational cells, since the numerical method used by the code tends to become extremely inefficient from a computational point of view when it is applied to problems that are larger and more complex.

The COBRA-IIIC/MIT code [9] is an improved version of

the original COBRA-IIIC code which takes advantage of the sparsity of the crossflow coefficient matrix and a more efficient coding scheme to reduce the running time and to permit the economic analysis of problems having more than 100 channels. Both COBRA-IIIC and COBRA-IIIC/MIT use the same physical and mathematical models to compute the flow and enthalpy distributions and have been found to give the same results for the classes of problems to which they can be applied.

Whereas all of the aforementioned codes were developed to model operational reactor conditions, the recently developed COBRA-IV-I [10] code represents significant progress in the field of thermal-hydraulic accident analysis because it solves the conservation equations as a true boundary value problem using an adaptation of the MAC method [11] to the control volume approach developed by Rowe [7]. The COBRA-IV-I code comprises all the features of a true benchmark code and can consider the effects of severe flow blockages, coolant expulsions, flow reversals, and recirculations. The code also contains a vastly improved version of the COBRA-IIIC code in which an iterative solution scheme for the crossflow distribution is used to reduce the running time to levels presumably similar to those obtained with COBRA-IIIC/MIT.

The THINC-IV code [2] is a Westinghouse proprietory, steady state thermal hydraulic analysis code which uses a velocity-perturbation theory method to compute the flow and

and enthalpy fields in reactor cores and fuel pin bundles in situations where the bundles and cores may contain flow blockages. The code uses the same placement of dependent variables in a computational cell as the MAC method [12] and does not appear to be applicable to problems in which the computational cells are allowed to have different sizes and geometrical shapes.

Perhaps the most advanced of all three-dimensional thermal-hydraulic analysis codes is the SABRE code [13] which is based upon the use of a primitive variable approach to solve for the steady-state temperature and velocity distributions in LMFBR fuel pin bundles. Both THINC-IV and SABRE have been verified extensively against experimental data, but little is known about running times of these codes or the precise limits of their range of applicability. A summary of the capabilities of the codes that have been discussed and a description of the methods upon which they are based is given in Table 1.1.

Based on these observations, the primary objectives of this thesis can be summarized as follows:

1. to develop more efficient numerical methods for solving the set of conservation equations used by the COBRA-IIIC and COBRA-IV codes during operational reactor conditions.

2. to incorporate these methods into a new computer code that has the capability of solving much larger and more complex problems than many other codes with much greater speed.

3. to lay a firm numerical basis for these methods and to show how they can be used to improve the representation and interpretation of the crossflow distributions.

4. to develop numerical methods which can be used for solving the fluid conservation equations by both direct and iterative techniques in a mixed lattice where the cells can have a variety of different sizes and geometrical shapes.

5. to compare these methods of solution to existing methods of solving the conservation equations in order to show how these methods have developed historically and to suggest ways by which the numerical procedures that have been applied to these methods can be improved.

Table 1.1. Summary of the Characteristics of Several Reactor Thermal-Hydraulic Analysis Codes

CODE	METHOD OF SOLUTION	REACTOR TYPES ANALYZED			COMPUTATIONS PERFORMED		GEOMETRIES ANALYZED
		BWR	PWR	LMFBR	STEADY STATE	TRANSIENT	
SUPERENERGY	Marching Method			X	X	X	Arbitrary
COBRA-II	Marching Method	X	X	X	X		Arbitrary
COBRA-IIIC	Pseudo Boundary Value Method	X	X	X	X	X	Arbitrary
COBRA-IIIC/ MIT	Pseudo Boundary Value Method	X	X		X	X	Arbitrary
COBRA-IIIP/ MIT	Pseudo Boundary Value Method	X	X		X	X	Arbitrary
THINC-IV	True Boundary Value Method		X		X		Subchannel only
COBRA-IV-I	True Boundary Value Method	X	X	X	X	X	Arbitrary
SABRE	True Boundary Value Method			X	X		Subchannel only

1.3 Mathematical Formulation

Due to the widespread acceptance and use of the COBRA-IIIC code, a considerable amount of the physical and mathematical formalism employed by the code will also be used as the basis for the numerical method to be developed here. This essentially means that the same basic set of governing conservation equations will be considered and that the same marching-type of solution procedure used for the initial value problem will be retained in space and time. The objectives of the methods to be proposed in this thesis are to solve the fluid conservation equations as efficiently as possible so that much larger and more complex problems can be handled within the limits of existing computational capacity.

1.4 Basic Assumptions

With these points in mind, the assumptions of primary importance to the following development are:

1. A lumped parameter approach is valid
2. Sonic velocity propagation is ignored
3. The diversion crossflow is usually smaller than the axial mass flow rate
4. Viscous dissipation is neglected
5. The liquid and vapor phases during boiling are in thermodynamic equilibrium
6. Electromagnetic body forces are ignored

7. Homogeneous two-phase flow may exist during boiling conditions
8. Flow reversals, recirculating flows, and coolant expulsions are not considered

CHAPTER II

2.1 Governing Set of Conservation Equations

By using the control volume approach described in reference [7], the equations for the conservation of mass, energy, and momentum of the fluid in a lattice composed of a number of computational cells can be written as

Continuity:

$$A_i \frac{\partial \rho_i}{\partial t} + \frac{\partial m_i}{\partial x} = - \sum_{j=1}^{N} w_{ij} \tag{1}$$

Energy:

$$\frac{1}{a_i''} \frac{\partial h_i}{\partial t} + \frac{\partial h_i}{\partial x} = \frac{q_i'}{m_i} - \sum_{j=1}^{N} (t_i - t_j) \frac{c_{ij}}{m_i} + \sum_{j=1}^{N} (h_i - h_j) \frac{w_{ij}}{m_i} + \sum_{j=1}^{N} (h_i - h^*) \frac{w_{ij}}{m_i} \tag{2}$$

Axial Momentum:

$$\frac{1}{A_i} \frac{\partial m_i}{\partial t} - 2u_i \frac{\partial \rho_i}{\partial t} + \frac{\partial P_i}{\partial x} = - \left(\frac{m_i}{A_i}\right)^2 \left[\frac{v_i f_i \phi_i}{2D_i} + \frac{k_i v_i}{2\Delta x} + A_i \frac{\partial}{\partial x}\left(\frac{v_i}{A_i}\right) \right] - g\rho_i \cos\theta - f_{wf} \sum_{j=1}^{N} (u_i - u_j) \frac{w_{ij}}{A_i} + \sum_{J=1}^{N} (2u_i - u^*) \frac{w_{ij}}{A_i} \tag{3}$$

Transverse Momentum:

$$\frac{\partial w_{ij}}{\partial t} + \frac{\partial}{\partial x}(u^* w_{ij}) + \left(\frac{s}{\ell}\right) C_{ij} w_{ij} = \left(\frac{s}{\ell}\right)(P_i - P_j) \tag{4}$$

where the summations indicated are to be performed over all cells (j = 1,2,. . . N) connected in the transverse plane to each cell represented by the index i. In this way, the differential equations for a typical cell in the hexagonal lattice shown in Fig. 2.1 (say cell 4) may be written explicitly by simply summing from j = 1 to 3 and the differential equations for cell 5 in the rectangular lattice shown in the figure may be generated in an analogous manner by summing from j = 1 to 4.

The equations for other layouts of cells, such as those in round bundles, are generated by simply summing over all the cells in the lattice that interact with one another in such a way that the interchange of mass, energy, and momentum is allowed to occur. As in COBRA-IIIC, this set of equations is closed by defining an additional equation for the physical state of the coolant of the form

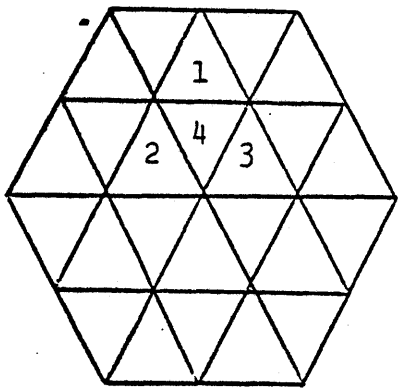
$$\rho_i = \rho(h_i, P^*) \qquad (5)$$

and empirical correlations for turbulent mixing coefficients, friction factors, heat transfer coefficients and other parameters used to characterize two-phase flows.

2.2. Matrix Form of the Conservation Equations

For a lattice composed of a large number of computational cells, the set conservation equations that has been

Hexagonal lattice



Rectangular lattice

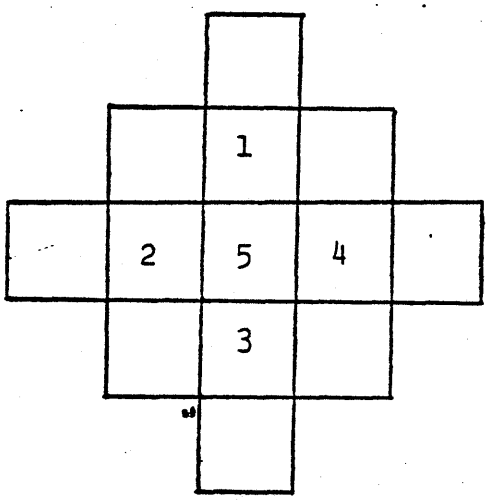


Fig. 2.1 Cell numbering schemes for generating systems of difference equations

discussed here becomes unwieldy; consequently, a more compact matrix notation is desirable. In reference [7] a vector form of the conservation equations is presented in which a transformation matrix [S] and its transpose $[S]^T$ is used to reduce these equations to a more tractable form. The elements of the matrix [S] are defined through two column vectors i(K) and j(K) where i and j represent the indices of adjacent computational cells and K is used to represent the index of the common boundary connecting them. As in COBRA-IIIC, the entries in [S] and $[S]^T$ may only have values of 0, 1, and -1, and the crossflow is arbitrarily considered to be positive when the dominant direction of the flow is from cell i to cell j, where i is less than j.

Using the definitions of [S] and $[S]^T$ presented in reference [7], the foregoing conservation equations may also be written as

Continuity:

$$[A]\left\{\frac{\partial \rho}{\partial t}\right\} + \left\{\frac{\partial m}{\partial x}\right\} = -[S]^T\{w\} , \qquad (6)$$

Energy:

$$\left\{\frac{1}{u''}\frac{\partial h}{\partial t}\right\} + \left\{\frac{\partial h}{\partial x}\right\} = \left[\frac{1}{m}\right]\{Q\} - \left[\frac{1}{m}\right][S]^T[\Delta h]\{w'\}$$

$$- \left[\frac{1}{m}\right][S]^T[\Delta t]\{c\} + \left[\frac{1}{m}\right]\left[[h][S]^T - [S]^T[h^*]\right]\{w\}, \qquad (7)$$

Axial Momentum:

$$[\tfrac{1}{A}]\ \{\tfrac{\partial \rho}{\partial t}\} - \{2u\ \tfrac{\partial \rho}{\partial t}\} + \{\tfrac{\partial P}{\partial x}\} = \{a'\} + \qquad (8)$$

$$[\tfrac{1}{A}]\ \left[[2u][S]^T - [S]^T[u^*]\right]\ \{w\}\ ,$$

where the components of the pressure drop due to frictional forces, gravitational forces, and the effects of turbulent mixing are given by

$$\{a'\} = -\left\{ (\tfrac{m}{A})^2 \left(\frac{vf\phi}{2D} + \frac{Kv'}{2\Delta x} + A\ \frac{\partial (v'/A)}{\partial x}\right) + \rho g\ \cos\phi \right\}$$

$$- f_t[\tfrac{1}{A}][S]^T[\Delta u]\{w'\}\ . \qquad (9)$$

and transverse interactions are taken into account by means of the matrix form of the transverse momentum equation developed for the COBRA-IIIC code:

Transverse Momentum:

$$\{\frac{\partial w}{\partial t}\} + \{\frac{\partial (u^* w)}{\partial x}\} + (\tfrac{s}{\ell})[C]\{w\} = (\tfrac{s}{\ell})[S]\ \{P\} \qquad (10)$$

For convenience, the same nomenclature and notation used in [7] are also used here. Consequently, this reference should be consulted for more detailed information about the definition and derivation of the individual terms.

2.3 Conversion of the Conservation Equations into Difference Equations

The preceding conservation equations must be converted into a system of difference equations before they can be solved by numerical means. Taking a first order backward differencing scheme for the spatial and temporal derivatives to provide numerical stability allows the equations for the conservation of mass and axial momentum to be written as

Continuity

$$[A_j]\left\{\frac{\rho_j - \bar{\rho}_j}{\Delta t}\right\} + \left\{\frac{m_j - m_{j-1}}{\Delta x}\right\} = -[S]^T\{w_j\} , \qquad (11)$$

Axial Momentum:

$$[A_j]^{-1}\left\{\frac{m_j-\bar{m}_j}{\Delta t}\right\} - [2u_j]\left\{\frac{\rho_j-\bar{\rho}_j}{\Delta t}\right\} + \left\{\frac{P_j-P_{j-1}}{\Delta x}\right\} =$$

$$\{a'_{j-1}\} + [A_j]^{-1}\left[[2u_j][S]^T - [S]^T[u^*_j]\right]\{w_j\} . \qquad (12)$$

For reasons to be discussed in section 2.7, the energy equation is approximated using a differencing scheme which is spatially explicit and temporally implicit:

Energy:

$$\left\{\frac{1}{u_j''}\right\}\left\{\frac{h_j-\bar{h}_j}{\Delta t}\right\} + \left\{\frac{h_j-h_{j-1}}{\Delta x}\right\} = \{m_j\}^{-1}\left\{\{Q_{j-1/2}\}-[S]\right.$$

$$\left.[\Delta h_{j-1}]\{w'_{j-1}\} - [S]^T[\Delta t_{j-1}][c_{j-1}]\right\} + \{m_j\}^{-1}$$

$$\left[[h_{j-1}][S]^T-[S]^T[h^*_{j-1}]\right]. \qquad (13)$$

In this differencing scheme (Fig. 2.2) each control volume is bounded by two adjacent control volumes in the axial direction and the interchange of mass, energy, and momentum is allowed to occur between adjacent computational cells in the transverse plane. A Taylor series expansion of Equations (11), (12) and (13) shows that this differencing scheme has a truncation error of $0(\Delta x, \Delta t)$. Although many other differencing schemes are possible, this scheme is the only one that will be considered here. Finally, it must be understood that the transverse momentum equation has been excluded from the preceeding discussion. This equation will now be considered in more detail since a better knowledge of its structural properties as well as the differencing schemes that can be applied to it are necessary to gain a more comprehensive understanding of the methods to be proposed.

2.4 Development of a New Spatial Differencing Scheme for the Transverse Momentum Equation

The purpose of the transverse momentum equation is to couple the computational cells together so that pressure gradients generated by the axial momentum equation can be used as the driving forces for the transport of mass, energy, and momentum between computational cells in the transverse

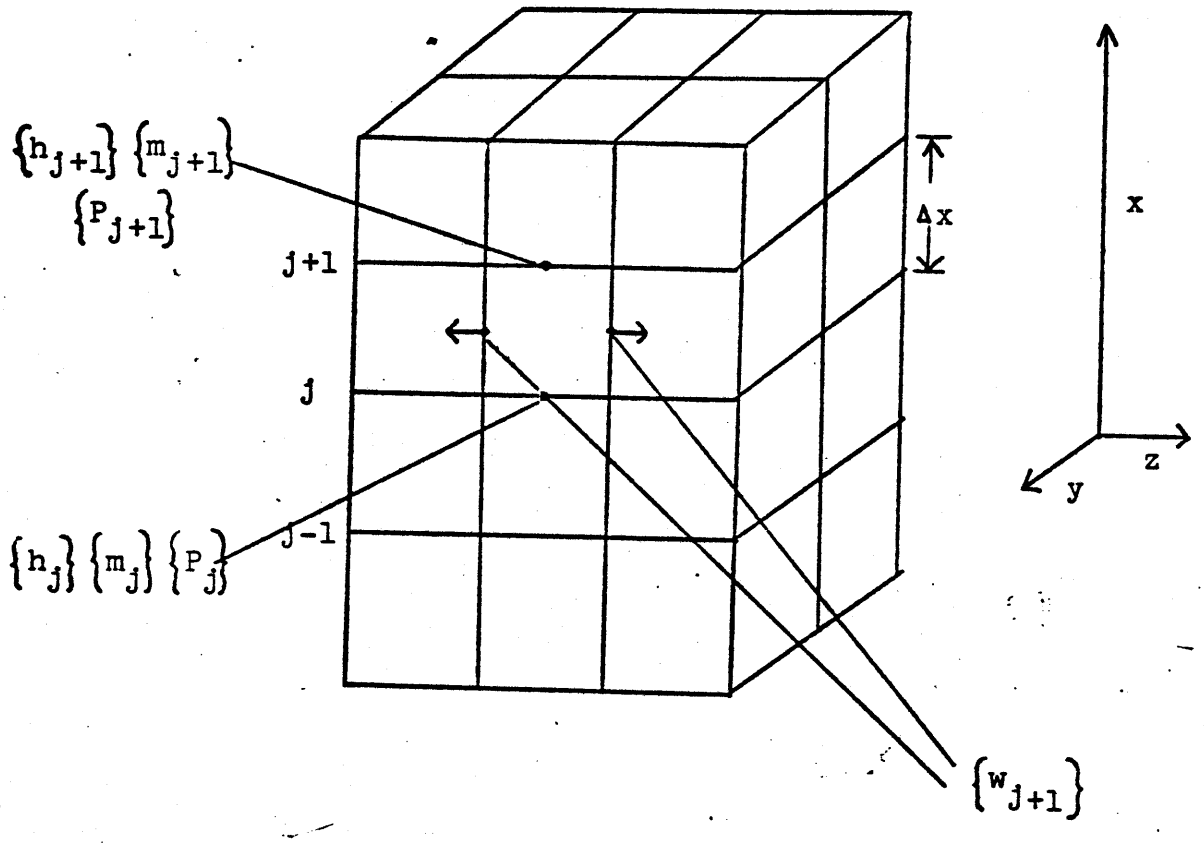


Fig. 2.2 Sample mesh grid to be used for numerical calculations

plane. The purpose of this section is to present a spatial differencing scheme for the transverse momentum equation which will be used as the basis for the numerical methods to be developed. The idea of the differencing scheme is to split the pressure vector and all the other transverse friction terms which drive the cross flow in the transverse momentum equation into a sum of spatially implicit and explicit parts. By introducing the weighting function, θ, having an arbitrary value between 0.0 and 1.0, the pressure vector $\{P\}$ and the cross flow resistance terms $[C]\{w\}$ in Equation (10) can be written as

$$\{P\} = \theta\{P_j\} + (1-\theta)\{P_{j-1}\} \tag{14}$$

$$[C]\{w\} = \theta[C_j]\{w_j\} + (1-\theta)[C_{j-1}]\{w_{j-1}\}, \tag{15}$$

where $0.0 \leq \theta \leq 1.0$.

As a result, it is then possible to write the transverse momentum equation more generally as

$$\left\{\frac{w_j-\bar{w}_j}{\Delta t}\right\} + \left\{\frac{u^*_j w_j - u^*_{j-1} w_{j-1}}{\Delta x}\right\} + \left(\frac{s}{\ell}\right)\left\{\theta[C_j]\{w_j\}+(1-\theta)[C_{j-1}]\{w_{j-1}\}\right\} = \left(\frac{s}{\ell}\right)[S]\left\{\theta\{P_j\}+(1-\theta)\{P_{j-1}\}\right\}, \tag{16}$$

where the significance of the proposed differencing scheme can be seen by examining the form of the transverse momentum equation for various values of θ; e.g., by setting θ equal to 0.0, 1/2, and 1.0 respectively,

$\underline{\theta = 0.0}$:

$$\left\{\frac{w_j-\bar{w}_j}{\Delta t}\right\}+\left\{\frac{u^*_j w_j-u^*_{j-1}w_{j-1}}{\Delta x}\right\}+\left(\frac{s}{\ell}\right)[C_{j-1}]\{w_{j-1}\} = \left(\frac{s}{\ell}\right)[S]\{p_{j-1}\} \tag{17}$$

$\underline{\theta = 1/2}$:

$$\left\{\frac{w_j-\bar{w}_j}{\Delta t}\right\}+\left\{\frac{u^*_j w_j - u^*_{j-1}w_{j-1}}{\Delta x}\right\}+\left(\frac{s}{\ell}\right)\left\{\frac{[C_j]\{w_j\}+[C_{j-1}]\{w_{j-1}\}}{2}\right\} = \left(\frac{s}{\ell}\right)[S]\left\{\frac{\{P_j\}+\{P_{j-1}\}}{2}\right\}, \tag{18}$$

$\theta = 1.0$:

$$\left\{\frac{w_j-\bar{w}_j}{\Delta t}\right\}+\left\{\frac{u^*_j w_j - u^*_{j-1}w_{j-1}}{\Delta x}\right\}+\left(\frac{s}{\ell}\right)[C_j]\{w_j\} = \left(\frac{s}{\ell}\right)[S]\{P_j\}. \tag{19}$$

Thus, it car be seen that an approach has been developed which allows the cross flow distribution to be driven by any combination of the pressure fields that exist at the top and the bottom of each plane of computational cells. By choosing a value for θ, the pressure fields from adjacent axial levels can be blended together in a manner that allows the degree of coupling of the transverse and axial momentum equations to be a function of the problem being solved. This blending of the pressure fields tends to eliminate sudden changes in the transverse cross flow distribution caused by discontinuities resulting from the application of the correlations for the pressure gradients at the interface between flow regimes and allows more flexibility to be used in modeling the transfer of momentum between the computational cells.

2.5 Development of a Pressure-Based "Theta Method"

2.5.1 Derivation of the Difference Equations for the Pressure Field

Using the aforementioned difference equations, Eqns.(11) - (16) as the starting point for the development of the proposed method, it should be noted that it is desirable to construct a procedure that simultaneously guarantees the conservation of mass and momentum at each axial elevation of the core at each instant of time. This condition can be satisfied by combining the continuity equation, Equation (11), together with the axial momentum equation, Equation (12), to explicitly eliminate the time dependent density term, $\{\frac{\partial \rho}{\partial t}\}$. The result of this substitution is a set of matrix equations of the form

$$[A_j]^{-1}\left\{\frac{m_j-\bar{m}_j}{\Delta t}\right\} + [2u_j][A_j]^{-1}\left\{\frac{m_j-m_{j-1}}{\Delta x}\right\} + [2u_j][A_j]^{-1}[S]^T\{w_j\}$$

$$+\left\{\frac{P_j-P_{j-1}}{\Delta x}\right\} = \{a'_{j-1}\} + [A_j]^{-1}[2u_j][S]^T\{w_j\} -$$

$$[A_j]^{-1}[s]^T[u^*_j]\{w_j\}. \qquad (20)$$

Since $[A_j]$, $[A_j]^{-1}$, and $[u_j]$ are diagonal matrices, Equation (20) can also be written as

$$[A_j]^{-1}\left\{\frac{m_j-\bar{m}_j}{\Delta t}\right\} + [2u_j][A_j]^{-1}\left\{\frac{m_j-m_{j-1}}{\Delta x}\right\} + \left\{\frac{P_j-P_{j-1}}{\Delta x}\right\}$$

$$= \{a'_{j-1}\}\,[A_j]^{-1}[S]^T[u^*_j]\{w_j\} \quad . \qquad (21)$$

The transverse momentum equation, Equation (16), is then solved for the cross flow distribution as function of the pressure distribution giving

$$\left[\frac{1}{\Delta t} + \frac{u^*_j}{\Delta x} + \left(\frac{s}{\ell}\right)\theta[C_j]\right] \{w_j\} = \left\{\frac{\bar{w}_j}{\Delta t}\right\} + \left\{\frac{u^*_{j-1} w_{j-1}}{\Delta x}\right\}$$

$$+ \left(\frac{s}{\ell}\right)(\theta-1)[C_{j-1}]\{w_{j-1}\} + \left(\frac{s}{\ell}\right)[S]\left\{\theta\{P_j\} + (1-\theta)\{P_{j-1}\}\right\}. \quad (22)$$

This system of equations can also be written in the form

$$\{w_j\} = [D_j]^{-1}\left\{\left\{\frac{\bar{w}_j}{\Delta t}\right\} + \left\{\frac{u^*_{j-1} w_{j-1}}{\Delta x}\right\} + \left(\frac{s}{\ell}\right)(\theta-1)[C_{j-1}]\{w_{j-1}\}\right\}$$

$$+\left(\frac{s}{\ell}\right)[D_j]^{-1}[S]\left\{\theta\{P_j\} + (1-\theta)\{P_{j-1}\}\right\},$$

$$\text{where } [D_j]^{-1} = \left[\frac{1}{\Delta t} + \frac{u^*_j}{\Delta x} + \left(\frac{s}{\ell}\right)\theta[C_j]\right]^{-1}. \quad (24)$$

Substituting Equation (23) into Equation (21) to explicitly eliminate the cross flow distribution results in the following system of equations for the pressure field

$$[A_j]^{-1}\left\{\frac{m_j - m_j}{\Delta t}\right\} + [2u_j][A_j]^{-1}\left\{\frac{m_j - m_{j-1}}{\Delta x}\right\} + \left\{\frac{P_j - P_{j-1}}{\Delta x}\right\}$$

$$= \{a'_{j-1}\} - [A_j]^{-1}[S]^T[u^*_j][D_j]^{-1}\left\{\left\{\frac{\bar{w}_j}{\Delta t}\right\} + \left\{\frac{u^*_{j-1} w_{j-1}}{\Delta x}\right\}\right.$$

$$\left. + \left(\frac{s}{\ell}\right)(\theta-1)[C_{j-1}]\{w_{j-1}\}\right\} - \left(\frac{s}{\ell}\right)[A_j]^{-1}[S]^T[u^*_j][D_j]^{-1}[S]$$

$$\left\{\theta\{P_j\} + (1-\theta)\{P_{j-1}\}\right\}. \quad (25)$$

Additional algebraic manipulation allows these equations to be written in more compressed form as

$$[I + \theta M_j]\{P_j\} = [I - (1-\theta)M_j]\{P_{j-1}\} + \{b_j\} \quad (26)$$

$$\text{where } [M_j] = \Delta x\left(\frac{s}{\ell}\right)[A_j]^{-1}[S]^T[u^*_j][D_j]^{-1}[S], \quad (27)$$

[I] is the identity matrix, and the vector $\{b_j\}$ is given by

$$\{b_j\} = \Delta x\{a'_{j-1}\} - \left(\frac{\Delta x}{\Delta t}\right)[A_j]^{-1}\{m_j - \bar{m}_j\} - [A_j]^{-1}[2u_j]\{m_j - m_{j-1}\}$$

$$-\Delta x[A_j]^{-1}[S]^T[u^*_j][D_j]^{-1}[S]\left\{\left\{\frac{\bar{w}_j}{\Delta t}\right\}+\left\{\frac{u^*_{j-1}w_{j-1}}{\Delta x}\right\}+\left(\frac{s}{\ell}\right)(\theta-1)[C_{j-1}]\{w_{j-1}\}\right\}. \quad (28)$$

Note that $\{b_j\}$ is a source vector containing terms which contribute to the axial pressure gradient as the result of frictional forces, gravitational forces, and the spatial and temporal acceleration of the flow. In spite of the seemingly complex structure of this vector, it should be noted that it contains only one entry from each computational cell. Finally, the value of $\{m_j\}$ in Equation (25) is unknown and therefore must be initially estimated and updated through iteration.

2.5.2 Discussion of the Difference Equations for the Pressure Field

In order to more clearly comprehend the consequences of the procedure that has been proposed, it is helpful to write Equation (26) as

$$\{P_j\} = [I+\theta M_j]^{-1}[I-(1-\theta)M_j]\{P_{j-1}\} + [I+\theta M_j]^{-1}\{b_j\} \quad (29)$$

and to examine the structure of the matrix equations governing the pressure field $\{P_j\}$ for various values of θ. Setting $\theta = 0.0$, 1/2, and 1.0 allows Equation (29) to be written in the following three forms

$\underline{\theta = 0.0}$

$$\{P_j\} = [I-M_j]\{P_{j-1}\} + \{b_j\} \quad (30)$$

$\underline{\theta = 1/2}$

$$\{P_j\} = \left[I+\frac{M_j}{2}\right]^{-1}\left[I-\frac{M_j}{2}\right]\{P_{j-1}\} + \left[I+\frac{M_j}{2}\right]^{-1}\{b_j\} \quad , \qquad (31)$$

$\underline{\theta = 1.0}$

$$\{P_j\} = [I+M_j]^{-1}\{P_{j-1}\} + [I+M_j]^{-1}\{b_j\} \quad . \qquad (32)$$

As discussed in section 2.4, these equations demonstrate that a direct relationship exists between the form of the equations for the pressure field and the forces that are used to drive the cross flow distribution between the computational cells in the transverse plane. Setting $\theta = 0.0$ allows the cross flow distribution to be driven by the pressure field that exists at the bottom of each plane of computational cells. From Equation (30) it can be seen that this results in a differencing scheme which is spatially explicit but temporally implicit. Choosing $\theta = 1.0$ means that the cross flow distribution is governed by the pressure field which exists at the top of each plane of computational cells. This in turn is numerically equivalent to requiring that the system of equations to be solved for the pressure field is fully implicit as denoted by Equation (32). Finally, by setting $\theta = 1/2$, Equation (31) shows that it is possible to generate a system of equations where the cross flow distribution is driven by the average of the pressure distributions that exist at the top and the bottom of each plane of computational cells. In this specific case, the difference equations are temporally implicit but have a

spatial component whose structure is analogous to that of the Crank-Nicholson method. It should be noted that other values of θ lead to other spatial differencing schemes, and in fact, a whole spectrum of these schemes can be generated from Equation (29) by selecting other values of θ. Finally, it should be recognized that reducing the value of θ increases the diagonal dominance of the coefficient matrix, but does so at the cost of reduced numerical stability (Section 6.3).

2.5.3 Derivation of the Cross Flow Distribution from the Pressure Field

When transverse pressure gradients develop between the computational cells at a given axial elevation, the flow field in the lattice becomes perturbed and a cross flow distribution is set up in order to equalize imbalances in the radial pressure field. To account for the effects of the cross flow distribution on the interchange of mass and momentum between the computational cells, the cross flow distribution must first be found from the transverse momentum equation, Equation (10). The pressure vector used to drive the cross flow distribution in this equation is found by blending the pressure field computed from Equation (29) together with the pressure field from the previous axial level to form the composite pressure field given by Equation (14) as the numerical

scheme sweeps downstream. The cross flow distribution at successive axial steps is then found from Equation (16) by rewriting it as

$$\{w_j\} = [D_j]^{-1}\left\{ \frac{\bar{w}_j}{\Delta t} + \frac{u^*_j w_{j-1}}{\Delta x} + \left(\frac{s}{\ell}\right)(\theta-1)[C_{j-1}]\{w_{j-1}\} \right\}$$

$$+ \left(\frac{s}{\ell}\right)[D_j]^{-1}[S]\left\{ \theta\{P_j\}+(1-\theta)\{P_{j-1}\} \right\}, \qquad (33)$$

where $[D_j]^{-1}$ is the diagonal matrix defined by Equation (24). This additional step requires very little computational effort since the inverse of $[D_j]$ can be found by inspection. Finally, it should be recognized that no assumption has been made so far with regard to θ; i.e., θ can still assume an arbitrary value between 0.0 and 1.0.

2.6 A Comparison of the "Theta Method" to the Numerical Method Used by COBRA-IIIC

In order to more clearly comprehend the capabilities of the method that has been developed, it is instructive to compare its features to those of the method used by COBRA-IIIC.

The starting point for the comparisom is the equation used to compute the cross flow distribution in the COBRA-IIIC code

$$[M_j]_c\{w_j\}_c = \{b_j\}_c \quad , \qquad (34)$$

where

$$[M_j]_c = \left[\frac{1}{\Delta t}\right] + \left[\frac{u^*_j}{\Delta x}\right] + \left(\frac{s}{\ell}\right)[C_j] +$$

$$\Delta x\left(\frac{s}{\ell}\right)[S][A_j]^{-1}\left[[2u_j][S]^T - [S]^T[u^*_j]\right] \qquad (35)$$

is the coefficient matrix that determines the transverse flow field,

$$\{b_j\}_c = \left\{\frac{\bar{w}_j}{\Delta t}\right\} + \left\{\frac{u^*_{j-1}w_{j-1}}{\Delta x}\right\} + \Delta x\left(\frac{s}{\ell}\right)[S]\left\{\{a'_j\} - [A_j]^{-1}\left\{\frac{m_j-\bar{m}_j}{\Delta t}\right\} + [2u_j]\left\{\frac{\rho_j-\bar{\rho}_j}{\Delta t}\right\}\right\} \qquad (36)$$

is the source vector that drives the transverse flow field, and the subscript c refers to the fact that these equations are used by the COBRA-IIIC code.

The comparison of the two methods is begun by examining the structure of the coefficient matrix used by each method and the consequence of these structural properties from a numerical point of view. A more detailed discussion of the meaning of the terms to be used is given in [7].

The analysis of Equation (34) starts by realizing that $[A_j]$, $[u_j]$, and $[u^*_j]$ are diagonal matrices. With this observation it is possible to write the cross flow coefficient matrix more generally as

$$[M_j]_c = [C_1][I] + [C_2][S][S]^T \quad , \qquad (37)$$

where $[C_1]$ and $[C_2]$ are matrices containing parameters that are problem dependent, [I] is the identity matrix, and [S] and $[S]^T$ are "interface or gap connection matrices" whose purpose is to maintain a consistent ordering scheme between the cells used to define the topology of the lattice and boundaries used to define interactions between adjacent cells.

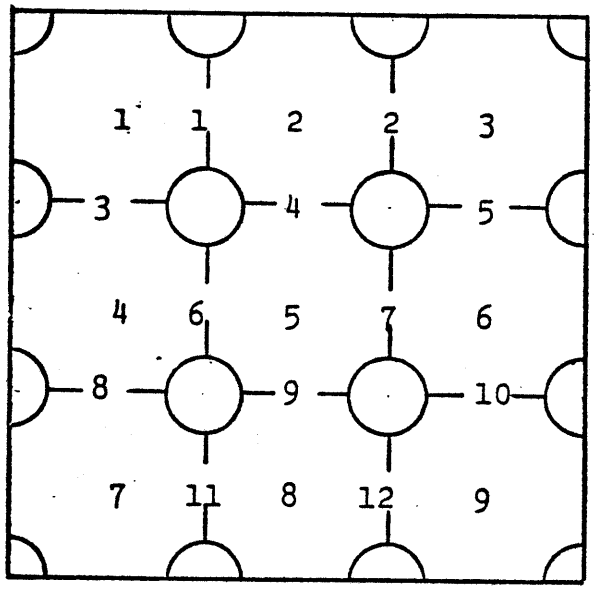


Fig. 2.3 A simple subchannel numbering scheme

Equation (37) shows that both the size and structure of the cross flow coefficient matrix are determined primarily by the order of multiplication of [S] and $[S]^T$ for the simple subchannel numbering scheme shown in Fig. 2.3.

In these equations, the number of cross flows is indicated by the symbol NF and the number of nodes at which the pressure is to be found by the symbol NP. Here, as in COBRA-IIIC, the cross flow is considered to be positive when the dominant direction of fluid flow is from cell i to cell j, where i is less than j. From Equations (38) and (39) it can be seen that [S] and $[S]^T$ are generally rectangular matrices rather than square matrices since they represent a topological situation in which there are considerably more boundaries than computational cells:

$$[S] = \overset{\longleftarrow\ NP\ \longrightarrow}{\begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}} \begin{matrix} \uparrow \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \downarrow \end{matrix} \begin{matrix} \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \end{matrix} \overset{NF}{} \qquad (38)$$

$$[S]^T = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & -1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 \end{bmatrix} \quad \begin{matrix} \leftarrow NF \rightarrow \\ \\ \\ \\ \\ \\ \\ \\ \\ \uparrow \\ NP \\ \downarrow \end{matrix} \qquad (39)$$

Thus [S] is an NPxNF matrix whereas its transpose, $[S]^T$, is an NFxNP matrix. Taking the product of [S] and $[S]^T$ for the subchannel numbering scheme shown in Fig. 2.3 results in an NFxNF matrix of the form

$$[S][S]^T = \begin{bmatrix} 2 & -1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 2 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 2 & 0 & 0 & -1 & 0 & -1 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 2 & 0 & 1 & -1 & 0 & -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 2 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 2 & -1 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & -1 & 2 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 2 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 & 0 & -1 & 1 & 0 & 2 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & -1 & 0 & -1 & 0 & 0 & 2 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 2 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & -1 & 2 \end{bmatrix} \quad \begin{matrix} \leftarrow NF \rightarrow \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \uparrow \\ NF \\ \downarrow \end{matrix} \qquad (40)$$

It should be noted that the structure of this coefficient matrix is very similar to the structure of the cross flow

coefficient matrix contained in the COBRA-IIIC code. The primary difference is that the coefficient matrix used by the code is modified by the addition of the diagonal terms in Equation (35):

$$[\frac{1}{\Delta t}] + [\frac{u^*_j}{\Delta x}] + (\frac{s}{\ell})[C_j] \qquad (41)$$

This modification generally improves the numerical properties of the matrix, and in many cases, appears to provide the diagonal dominance necessary to guarantee the success of iterative solution techniques. However, the iteration is relatively slow to converge, and is not much more efficient from a computational point of view than Gaussian elimination, particularly for problems where the axial mesh spacing is very large (i.e., approximately equal to one foot). For this reason, the cross flow distribution is found in COBRA-IIIC by solving Eq. (34) by Gaussian elimination, and it is the repeated application of this inversion technique to Eq. (34) at each axial level of the core at each instant of time that is primarily responsible for the long running time of the code.

For the proposed method, the coefficient matrix governing the pressure distribution can be written as

$$[I + \theta M_j] \qquad (42)$$

Since [I]is the identity matrix and $[M_j]$ is defined by Equation (27), the size and structure of this coefficient matrix is determined primarily by the order of multiplication

of [S] and $[S]^T$. Taking the expressions for $[S]^T$ and [S] given previously and multiplying them together in the proposed order gives

$$[S]^T[S] = \begin{bmatrix} 2 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 3 & -1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & 0 & 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 3 & -1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & -1 & 4 & -1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & -1 & 3 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 & 0 & 0 & 2 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & -1 & 3 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 2 \end{bmatrix} \begin{matrix} \\ \\ \\ \\ \\ \\ \\ \\ \end{matrix} \quad \begin{matrix} \leftarrow \text{NP} \rightarrow \\ \\ \\ \\ \\ \\ \\ \\ \\ \end{matrix} \quad \begin{matrix} \uparrow \\ \\ \\ \\ \\ \text{NP} \\ \\ \\ \downarrow \end{matrix} \tag{43}$$

Thus it can be seen that the coefficient matrix governing the pressure field has a structure which suggests that the radial pressure distribution obeys a type of Poission equation. This is a consequence of the fact that the solution scheme is formulated to take advantage of the "primitive variables" governing the flow field, and one of them is the pressure $\{p_j\}$. Furthermore, since the diagonal entries in Equation (43) are equal in magnitude and opposite in sign to the sum of the off-diagonal entries in any row or column, the matrix has a much simpler and more predictable band structure than that of Equation (40). This numerically desirable feature stems from the fact that each computational cell in the transverse plane at which the pressure is to be

found is only connected to its nearest neighboring cells, whereas in COBRA-IIIC, the crossflow across one boundary may be affected by the crossflow across as many as six other boundaries in the lattice of cells shown in Fig.2.3. Thus it can be seen that the band width of the coefficient matrix is reduced by approximately a factor of two if a solution scheme based upon the pressure is used.

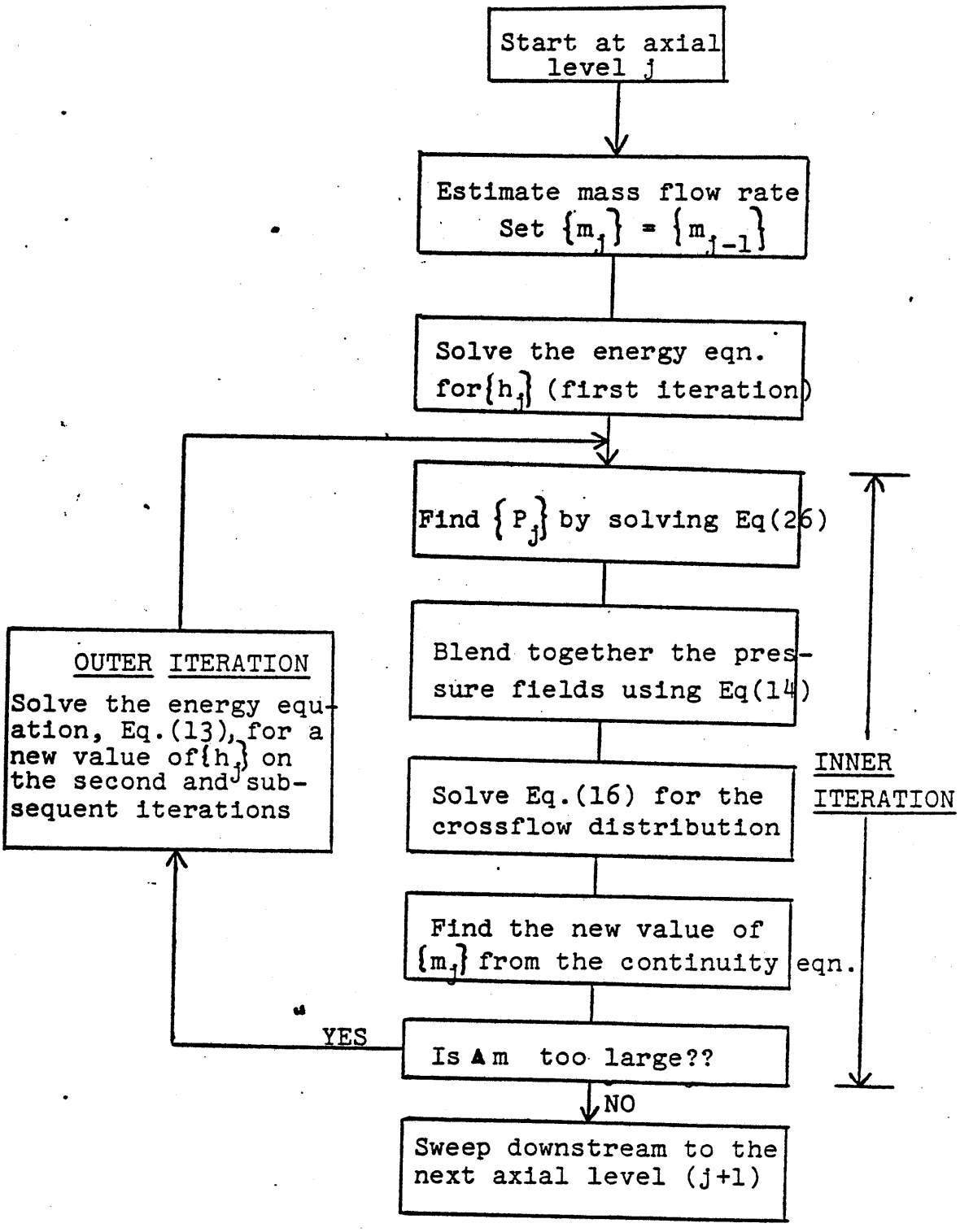
The primary computational advantage of the proposed method lies in the fact that the coefficient matrix governing the pressure field, Equation (42), is a Stieltjes matrix; that is to say, the matrix is a diagonally dominent, irreducible, positive definite matrix with a simple and predictable band structure for any non-trivial space step and time step size. For consistent numbering schemes, the convergence of standard iterative solution techniques for matrices of this type can be guaranteed. Consequently, it is now possible and advisable to find the pressure distribution in the lattice by iterative techniques rather than by the method of Gaussian elimination. In Chapter 5 it will be shown that the use of an iterative solution scheme is often more efficient than a direct solution scheme, and can save a considerable amount of time and effort in the computation of the pressure distribution.

2.7 Use of the Conservation of Energy Equation with the Theta Method

The purpose of the preceding discussion has been to develop a general numerical method for solving the equations for the conservation of mass and momentum in fuel pin bundles

and nuclear reactor cores. The energy equation,Equation (2), was not considered explicitly because an understanding of its structure was not necessary for the development of the proposed method. However, after the equations for the conservation of mass and momentum have been solved, the thermal response of the fluid must be interfaced with the hydraulics and the energy transferred from the pins through heat transfer coefficients. This is generally done by solving the energy equation, Equation (13), for the enthalpy of the fluid at a new axial level, h_j, using the value of the enthalpy from the previous axial level, h_{j-1}, and the rate of energy generation known to exist in each computational cell. The energy balance is performed using the axial mass flow rates, $\{m_j\}$, found from a previous solution of the momentum equations. Thus the solution of the energy equation can be looked upon as an "outer iteration" in the two-step solution procedure. Generally speaking, it would be necessary to iterate between the energy equation and the momentum equations at least once at each axial elevation to obtain a self-consistent solution to the entire system of conservation equations. This is especially necessary for problems that involve considerable departures from operational reactor conditions (i.e., moderately severe transients). The relationship between the "outer iteration" scheme for the energy equation and the "inner iteration" scheme for the momentum equations is illustrated in the flow chart in Figure 2.4.

Fig. 2.4 Flow Chart for Iterations Between the Energy Equation and the Momentum Equations



2.8 Boundary Conditions for the Theta Method

Since the theta method may be used to solve an initial value problem in which the flow and enthalpy fields are found by marching downstream between the inlet and the outlet of a reactor core or rod bundle, it is necessary to specify the values of the axial mass flow rates, the pressures, and the enthalpies at the inlet of each channel to obtain a self-consistent solution to the entire system of conservation equations. Alternatively, it is possible for the method to be formulated as a "boundary value approach" in which the exit pressure may be specified and the inlet pressure level is "floated" to give a uniform outlet pressure. In either case, the sequential nature of the solution procedure allows arbitrary spatial and temporal forcing functions to be applied to the system pressure, the inlet enthalpy, the inlet mass flow rate, and the radial and axial power distributions. The crossflow at the core inlet is usually assumed to be zero, although more detailed crossflow distributions may be prescribed if a better physical picture is required and enough experimental evidence is available to specify it. The axial iteration schemes that may be applied to deal with the difference equations used by the theta method are discussed in more detail in Chapter 4.

2.9 Drawbacks of the Theta Method for a Flow-Blockage Type of Analysis

Whereas the coefficient matrix governing the pressure distribution is a Stieltjes matrix for cases in which the theta method is used to solve for the pressure at level j as a function of the pressure at level j-1 (see Eq. 26), the method has the drawback that it cannot be applied rigorously to problems where the crossflow is very large because the terms in the axial momentum equation which account for frictional and gravitational effects are evaluated at the bottom of a node rather than at the top of a node by means of a relationship of the form

$$\{a_{j-1}'\} = \{k_j\}\{m_{j-1}^2\} - \{f_j\} \tag{44}$$

This approximation is acceptable as long as the change in the axial mass flow rate, $\Delta m = \{m_j - m_{j-1}\}$ between the top and the bottom of a node is small, since an order of magnitude analysis presented in greater detail in section 3.1 shows that

$$\begin{aligned}\{a_j'\} &= -\{k_j m_j^2\} - \{f_j\} \\ &= -\{k_j\}\{m_{j-1}+\Delta m\}^2 - \{f_j\} \\ &= -\{k_j\}\{m_{j-1}^2\} - \{k_j\}\{2m_{j-1}\Delta m + \Delta m^2\} - \{f_j\}\end{aligned} \tag{45}$$

$$\text{or} \quad \{a_j'\} \simeq -\{k_j\}\{m_{j-1}^2\} - \{f_j\} \simeq \{a_{j-1}'\} \tag{46}$$

$$\text{if} \quad \Delta m << \{m_{j-1}\}$$

However, in the event of a flow blockage, Δm can be almost as large as $\{m_{j-1}\}$, and the $\{a_{j-1}'\} = \{a_j'\}$ approximation cannot be justified a priori. To rectify this limitation of the theta method, a more general numerical method called the MAT method (Modified and Advanced Theta Method) will now be developed. Like the original theta method, the MAT method also uses the concept of splitting the forcing terms and the pressure fields which drive the crossflow distribution in the transverse momentum equation into a sum of spatially implicit and explicit parts. However, because of the fact that the MAT method is based upon evaluating the terms which account for the frictional and gravitational effects directly at axial level j, the method is inherently more stable and can consider problems in which the crossflow is even larger than the axial mass flow rate. This feature of the method gives it the capability to analyze flow blockages and other types of problems that involve severe departures from operational reactor conditions. The numerical basis of the MAT method will be discussed in detail in the presentation to follow.

CHAPTER III

3.1 The MAT Method

3.1.1 Derivation of the Difference Equations for the MAT Method

If the continuity equation, Eq. (11), given in section 2.3 is written as

$$\{m_j\} = \{m_{j-1}\} - \Delta x[S]^T\{w_j\} - [A_j]\{\frac{\rho_j - \bar{\rho}_j}{\Delta t}\}\Delta x \tag{47}$$

it is possible to express the axial mass flow rate at level j in terms of the axial mass flow rate at level j-1 by means of a relationship of the form

$$\{m_j\} = \{m_{j-1}\} + \{\Delta m\} \tag{48}$$

where $\{\Delta m\} = -\Delta x[S]^T\{w_j\} - \Delta x[A_j]\{\frac{\rho_j - \bar{\rho}_j}{\Delta t}\}$ (49)

is the change in the axial mass flow rate between the top and the bottom of each plane of computational cells.

Using this relationship, the terms in the axial momentum equation which account for the pressure drop due to frictional and gravitational effects can be written as

$$\{a_j'\} = -\{k_j\}\{m_j^2\} - \{f_j\} \tag{50}$$

$$= -\{k_j\}\{m_{j-1} + \Delta m\}^2 - \{f_j\} \tag{51}$$

$$= -\{k_j\}\{m_{j-1}^2 + 2m_{j-1}\Delta m + \Delta m^2\} - \{f_j\} \tag{52}$$

where the $\{\Delta m^2\}$ term is generally small enough to be discarded for operational reactor conditions. However, because of the

possibility that $\{\Delta m^2\}$ may be as large as $\{m_{j-1}\}^2$ for cases in which there is a severe flow blockage, an order of magnitude analysis* shows that it may not be appropriate to construct a system of equations in which the effects of the presence of the $\{\Delta m\}^2$ term are ignored. For the sake of numerical convenience, the flow-squared terms are written in the form suggested by Rowe[7]:

---

*Order of Magnitude Analysis

If Δm is very small compared to m_{j-1}, Eq. (52) can be written as

$$\underset{1}{\{m_{j-1}\}}^2 + \underset{1\ \delta}{\{2m_{j-1}\Delta m\}} + \underset{\delta^2}{\{\Delta m\}^2} \simeq \{m_{j-1}\}^2$$

and when Δm is somewhat larger,

$$\underset{1}{\{m_{j-1}\}}^2 + \underset{1\ \delta}{\{2m_{j-1}\Delta m\}} + \underset{\delta^2}{\{\Delta m\}^2} \simeq \{m_{j-1}\}^2 + \{2m_{j-1}\Delta m\}$$

Note that very large values of Δm require that

$$\underset{1}{\{m_{j-1}\}}^2 + \underset{1\ 1}{\{2m_{j-1}\Delta m\}} + \underset{1^2}{\{\Delta m^2\}} = \{m_{j-1}\}^2 + \{2m_{j-1}+\Delta m\}\Delta m$$

$$= \{m_{j-1}\}^2 + \{m_j+m_{j-1}\}\Delta m$$

$$\{m_j\}^2 = \{m_{j-1}\}^2 + \{2m_{j-1}+\Delta m\}\{\Delta m\} \tag{53}$$

and since

$$\{\Delta m\} = \{m_j\} - \{m_{j-1}\} \tag{54}$$

the $\{\Delta m\}^2$ term can be retained by writing $\{m_j\}^2$ as

$$\{m_j^2\} = \{m_{j-1}\}^2 + \{m_j+m_{j-1}\}\{\Delta m\} \tag{55}$$

The terms which account for the pressure drop due to frictional and gravitational effects may then be expressed as

$$\{a_j'\} = -\{k_j\}\{m_{j-1}\}^2 - \{k_j\}\{m_j+m_{j-1}\}\Delta m - \{f_j\} \tag{56}$$

Applying the same reasoning to the temporal acceleration term in the axial momentum equation gives

$$\frac{[A_j]^{-1}}{\Delta t}\{m_j-\bar{m}_j\} = \frac{[A_j]^{-1}}{\Delta t}\{m_j-m_{j-1}+m_{j-1}-\bar{m}_j\}$$

$$= \frac{[A_j]^{-1}}{\Delta t}\{m_j-m_{j-1}\}+\frac{[A_j]^{-1}}{\Delta t}\{m_{j-1}-\bar{m}_j\} \tag{57}$$

and using the definition of Δm given by Eq. (48), it is possible to cast the temporal acceleration term into the alternative form

$$\frac{[A_j]^{-1}}{\Delta t}\{m_j-\bar{m}_j\} = [A_j]^{-1}\left\{\frac{\Delta m}{\Delta t}\right\} + [A_j]^{-1}\left\{\frac{m_{j-1}-\bar{m}_j}{\Delta t}\right\} \tag{58}$$

Substituting Eqs. (56) and (58) into the axial momentum equation given previously allows the conservation of momentum in the axial direction to be written as

$$\left\{\frac{P_j - P_{j-1}}{\Delta x}\right\} = \left\{2u_j \frac{\rho_j - \bar{\rho}_j}{\Delta t}\right\} + [A_j]^{-1} \left\{\frac{\bar{m}_j - m_{j-1}}{\Delta t}\right\}$$

$$-\{k_j\}\{m_{j-1}\}^2 - \{f_j\} - \left\{\frac{[A_j]^{-1}}{\Delta t} + \{k_j\}\{m_j + m_{j-1}\}\right\}\{\Delta m\}$$

$$+ [A_j]^{-1}\left[[2u_j][S]^T - [S]^T[u_j^*]\right]\{w_j\} \qquad (59)$$

and since the continuity equation, Eq. (49), requires that

$$\{\Delta m\} = -\Delta x[S]^T\{w_j\} - \Delta x[A_j]\left\{\frac{\rho_j - \bar{\rho}_j}{\Delta t}\right\} \qquad (60)$$

it is possible to simultaneously guarantee the conservation of mass and momentum at each axial level of the core at each instant of time by writing Eq. (59) as

$$\left\{\frac{P_j - P_{j-1}}{\Delta x}\right\} = [A_j]^{-1}\left\{\frac{\bar{m}_j - m_{j-1}}{\Delta t}\right\} - \{k_j\}\{m_{j-1}\}^2 - \{f_j\}$$

$$+ \left\{\left(2u_j + \frac{\Delta x}{\Delta t} + \Delta x k_j A_j (m_j + m_{j-1})\right)\left(\frac{\rho_j - \bar{\rho}_j}{\Delta t}\right)\right\}$$

$$+ [A_j]^{-1}\left[[2u_j + \frac{\Delta x}{\Delta t}][S]^T - [S]^T[u_j^*]\right]\{w_j\}$$

$$+ \Delta x[K_j(m_j + m_{j-1})]\ [S]^T\{w_j\} \qquad (61)$$

Solving the transverse momentum equation for the crossflow distribution as a function of the pressure distribution gives

$$\left[\frac{1}{\Delta t} + \frac{u_j^*}{\Delta x} + \left(\frac{s}{\ell}\right)\theta[C_j]\right]\{w_j\} = \left\{\frac{\bar{w}_j}{\Delta t}\right\} + \left\{\frac{u_{j-1}^* w_{j-1}}{\Delta x}\right\}$$

$$+ \left(\frac{s}{\ell}\right)(\theta - 1)[C_{j-1}]\{w_{j-1}\} + \left(\frac{s}{\ell}\right)[S]\left\{\theta\{P_j\} + (1-\theta)\{P_{j-1}\}\right\} \qquad (62)$$

This system of equations can also be written in the form

$$\{w_j\} = [D_j]^{-1}\{O_j\} + (\tfrac{s}{\ell})[D_j]^{-1}[S]\left\{\theta\{P_j\} - (1-\theta)\{P_{j-1}\}\right\} \quad (63)$$

where $[D_j]^{-1}$ is the diagonal matrix defined by

$$\left[\frac{1}{\Delta t} + \frac{u_j^*}{\Delta x} + (\tfrac{s}{\ell})\theta[C_j]\right]^{-1} \quad (64)$$

and the other forcing terms in the transverse momentum equation are represented by

$$\{O_j\} = \left\{\left\{\frac{\bar{w}_j}{\Delta t}\right\} + \left\{\frac{u_{j-1}^* w_{j-1}}{\Delta x}\right\} + (\tfrac{s}{\ell})(\theta-1)[C_{j-1}]\{w_{j-1}\}\right\} \quad (65)$$

Finally, substituting Eq. (63) into Eq. (61) to explicitly eliminate the crossflow distribution results in the following system of fundamental equations for the pressure field

$$\begin{aligned}\{P_j - P_{j-1}\} &= \frac{\Delta x}{\Delta t}[A_j]^{-1}\{\bar{m}_j - m_{j-1}\} - \Delta x\{k_j\}\{m_{j-1}^2\} - \Delta x\{f_j\} \\ &+ \frac{\Delta x}{\Delta t}\{\rho_j - \bar{\rho}_j\}\{2u_j + \frac{\Delta x}{\Delta t} + \Delta x k_j A_j (m_j + m_{j-1}) \\ &+ \Delta x[A_j]^{-1}\left[[2u_j + \frac{\Delta x}{\Delta t} + \Delta x[A_j]k_j(m_j+m_{j-1})[S]^T - [S]^T[u_j^*]\right] \\ &\left\{[D_j]^{-1}\{O_j\} + (\tfrac{s}{\ell})[D_j]^{-1}[S]\left\{\theta\{P_j\} + (1-\theta)\{P_{j-1}\}\right\}\right\}\end{aligned} \quad (66)$$

Additional algebraic manipulation allows these equations to be written in more compressed form as

$$[I + (1-\theta)M_j']\{p_{j-1}\} = [I - \theta M_j']\{p_j\} + \{b_j'\} \quad (67)$$

where I is the identity matrix,

$$\begin{aligned}[M_j'] = \Delta x(\tfrac{s}{\ell})&\left[[A_j]^{-1}[B_j][S]^T[D_j]^{-1}[S]\right. \\ &\left. - [A_j]^{-1}[S]^T[u_j^*][D_j]^{-1}[S]\right]\end{aligned} \quad (68)$$

is a coefficient matrix governing the pressure field,

$$\{b'_j\} = \frac{\Delta x}{\Delta t}[A_j]^{-1}\{\bar{m}_j - m_{j-1}\} - \Delta x\{k_j\}\{m_{j-1}^2\} - \Delta x\{f_j\}$$

$$+ \frac{\Delta x}{\Delta t}\{\rho_j - \bar{\rho}_j\}\{2u_j + \frac{\Delta x}{\Delta t} + \Delta x k_j A_j (m_j + m_{j-1})\}$$

$$+ \Delta x[A_j]^{-1}[B_j][S]^T[D_j]^{-1}\{O_j\} - \Delta x[A_j]^{-1}[S]^T[u_j^*][D_j]^{-1}\{O_j\} \quad (69)$$

is a source vector that contains terms which contribute to the axial pressure gradient, and

$$[B_j] = \left[[2u_j] + \frac{\Delta x}{\Delta t} + \Delta x[A_j]k_j(m_j + m_{j-1})\right] \quad (70)$$

is a matrix which is used in the construction of the source vector and the coefficient matrix governing the pressure field.

3.1.2 Discussion of the Difference Equations Used by the MAT Method

It is interesting to observe that by simply manipulating the difference equations into a different form before eliminating the crossflow distribution, it has been made possible to construct a system of equations for the pressure field in which the order of appearance of the weighting functions θ and $(1-\theta)$ have been reversed. As a result of the interchange of these weighting functions in the coefficient matrices operating on $\{p_j\}$ and $\{p_{j-1}\}$, a numerical procedure has been constructed in which a value of $\theta = 1.0$ corresponds to a differencing scheme which is spatially explicit, and a value of $\theta = 0.0$ corresponds to a differencing scheme which is spatially implicit. Setting $\theta = 1/2$, however, is still equivalent to requiring that the difference equations have a spatial component whose structure is analogous to that of the Crank-Nicholson method.

One of the primary computational advantages of the MAT method is the fact that the coefficient matrix operating on $\{p_{j-1}\}$ is a Stieltjes matrix, so that it is now possible to solve Eq. (67) for $\{p_{j-1}\}$ using an iterative solution technique such as the method of successive overrelaxation. Moreover, Eq. (68) shows that $[M_j']$ can be written as a sum of two matrices $[M_j]$ and $[M_j"]$:

$$[M_j'] = [M_j] + [M_j"] \tag{71}$$

where $$[M_j] = -\Delta x(\frac{s}{\ell})[A_j]^{-1}[S]^T[u_j^*][D_j]^{-1}[S] \tag{72}$$

is the coefficient matrix which appears in the equations of the original theta method, Eq. (27), and

$$[M_j"] = \Delta x(\frac{s}{\ell})[A_j]^{-1}[B_j][S]^T[D_j]^{-1}[S] \tag{73}$$

is a new coefficient matrix whose purpose is to modify the pressure field to account for the presence of the higher order terms, $\{2m_{j-1}\Delta m\}$ and $\{\Delta m^2\}$ in Eq. (52). Both $[M_j]$ and $[M_j"]$ are NPxNP matrices because their structure is determined primarily by the order of multiplication of $[S]^T$ and $[S]$. Similarly, it can be seen that the source vector $\{b_j'\}$, which is responsible for generating the axial pressure distribution, again contains only one entry from each computational cell.

3.2 Boundary Conditions for the MAT Method

The fact that the equations used by the MAT method are based upon solving for the pressure at axial level j-1 as a function of the pressure at axial level j makes it possible to develop a computational procedure in which it is not necessary to know the form of the inlet pressure distribution to find the flow and enthalpy fields. Instead, the exit pressure differentials between channels may be set equal to zero in the manner outlined in section 4.1.3 and the inlet pressure distribution may be "floated" to give a uniform outlet pressure. Thus the MAT method has been constructed to use exactly the same set of boundary conditions for the system pressure that are used by the crossflow approach in the COBRA-IIIC code. Since all the conservation equations used by the MAT method are to be solved sequentially rather than simultaneously, it is also necessary to specify the inlet enthalpy distribution, the inlet mass flow rates, and the inlet crossflow distribution. The crossflow at the core inlet is usually assumed to be zero, although more detailed crossflow distributions may be prescribed if a better physical picture is required and if there is enough experimental evidence available to specify it. The "boundary conditions" that are to be used with the methods discussed in this thesis are summarized in Table 3.1. Thus it can be seen that only the adaptation of the MAC method used by COBRA-IV-I,

Table 3.1. Boundary Conditions for Various Types of Solution Techniques

Type of Code or Method	Inlet Conditions	Outlet Conditions
COBRA-II (Initial Value Approach)	Specify p,m,w, and h	Outlet pressure distribution is floated
COBRA-IIIC (Boundary Value Approach)	Specify m,w, and h Inlet pressure distribution is "floated"	[S]{p}=0 (Outlet pressure is set equal to a constant)
THETA Method (Initial Value Approach)	Specify p,m,w, and h	Outlet pressure distribution is floated
THETA Method (Boundary Value Approach)	Specify m,w, and h Inlet pressure distribution is "floated"	[S]{P}=0 (Outlet pressure is set equal to a constant)
MAT Method (Initial Value Approach)	Specify p,m,w, and h	Outlet pressure distribution is floated
MAT Method (Boundary Value Approach)	Specify m,w, and h Inlet pressure distribution is "floated"	[S]{p}=0 (Outlet pressure is set equal to a constant)
COBRA-IV-I (True Boundary Value Approach)	Specify m,w, and p at the core inlet	Specify m,w, and p at the core outlet

which is a true boundary value method, requires a knowledge of the pressure distribution at both the inlet and the outlet of the core.

3.3 A Comparison of the Numerical Properties of the MAT Method and the Theta Method

Although the theta method has the desirable numerical property that it works with a Stieltjes matrix when it is used to solve for the pressure at axial level j as a function of the pressure at axial level j-1, the MAT method has the advantage that it is more suitable for solving the inverse of this problem (i.e., finding the pressure at axial level j-1 as a function of the pressure at axial level j) because the coefficient matrix

$$[I + (1-\theta)M_j'] \qquad (74)$$

used by the MAT method is a Stieltjes matrix for this case.

This difference between the properties of the methods is extremely significant from a numerical point of view. This is because the solution scheme in COBRA-IIIC attempts to drive the crossflow distribution at axial level j by the pressure differences evaluated with respect to axial level j-1, whereas the solution scheme used in COBRA-II is based on the inverse approach of driving the crossflow distribution by the pressure field at level j. Thus, the theta method is similar to the initial value approach used in COBRA-II (see section 4.1.2), and the MAT method more closely resembles

the "boundary value approach" used in COBRA-IIIC. However, because of the better approximation to the flow squared terms used by the MAT method, and the fact that it is not formulated as an initial value approach, the MAT method has been found to be considerably more stable (see Section 6.3) and can be applied rigorously to problems in which the diversion crossflow becomes even larger than the axial mass flow rate. Finally, it will be shown in the sections to follow that the MAT method can be extended very easily to deal with situations in which flow disturbances can be propagated several nodes upstream, whereas the theta method is not nearly as stable or as efficient when it is applied to problems of this type.

CHAPTER IV

4.1 Axial Iteration Schemes for the Theta Method and the MAT Method

In order to implement the numerical methods which have been developed in this thesis into a reactor thermal hydraulic analysis code, it is necessary to consider the types of axial iteration schemes that can be applied to the difference equations used by these methods. This requires an in-depth analysis of each of these schemes with respect to the physical assumptions they use, their computational efficiency, and their compatibility with different kinds of boundary conditions. Since the axial iteration schemes used by reactor thermal hydraulic analysis codes are generally classified as either (a) single pass schemes, (b) single pass schemes which are locally iterative, or (c) multipass schemes, the primary features of each of these schemes will now be considered separately.

4.1.1 Single Pass Axial Iteration Schemes

Single pass axial iteration schemes are really nothing more than marching techniques in which the initial conditions at the reactor inlet are given and in which a solution is obtained by marching downstream through the lattice in the axial direction only once. These schemes are extremely efficient from a computational point of view because they usually involve solving an explicit energy equation and

an implicit or explicit form of the momentum equations only once at each axial level of the core. However, they are generally considered to be the least accurate from a physical point of view because they do not contain a mechanism for correcting errors that may be propagated through the solution of the difference equations as the result of an inaccurate initial guess of either the pressure, the enthalpy, or the axial mass flow rate at each axial elevation of interest. For this reason, it is believed that a single pass axial iteration scheme is not the most appropriate scheme to be used with the methods that have been developed here, although it should be pointed out that well known codes such as CHIC-KIN [14] are based on single pass axial iteration schemes.

4.1.2 Single Pass Locally Iterative Axial Iteration Schemes

The next level of sophistication that may be introduced into a reactor thermal hydraulic analysis code is a single pass, locally iterative axial iteration scheme. This type of iteration scheme is based on the premise of sweeping downstream only once, but differs from a true marching technique because it performs additional iterations on the flow or pressure fields at specific axial planes in the core where these fields do not converge to a specified tolerance after the first iteration. Some iteration schemes of this type, such as those contained in HAMBO [6] and MIXER [15], attempt to equalize transverse pressure gradients at each axial

elevation in the core while the numerical scheme sweeps downstream, whereas the iterative scheme used in COBRA-II, for example, is slightly more realistic because it is based on the assumption that appreciable transverse pressure gradients can exist across a fuel pin bundle or a reactor core and that these pressure gradients are the primary driving force for the transverse crossflow distribution. Thus the techniques used in the HAMBO and MIXER codes would be expected to be applicable to closed bundles, where transverse pressure gradients are fairly small, whereas the crossflow approach used by the COBRA series of codes would be expected to be more appropriate for analyzing the "open lattice" found in a PWR core, where transverse pressure gradients can sometimes be significant. Generally speaking, locally iterative schemes are just slightly more expensive than single pass schemes because they require additional iterations to be performed only at those planes where the solution initially fails to converge, but these schemes are generally much more accurate from a physical point of view, particularly in problems where there is boiling, since they provide a mechanism for correcting errors in the initial estimates of the pressure fields and the flow fields at selected axial elevations in the core. The original version of the theta method and the numerical scheme used in COBRA-II are both examples of methods that

are suitable for use with a single pass, locally iterative axial iteration scheme. Although the continuity equation and the momentum equations used by the theta method are time dependent and are applicable to more complex situations than the equations used in COBRA-II (which are only able to handle steady state problems) the procedures applied to solve both sets of equations, as illustrated in the flow charts in Figs. 4.1 and 4.2, are essentially the same. After initial values of {p} , {h}, {w} , and {m} are established at the core inlet, a differencing scheme is constructed in which an energy equation is solved explicitly for the enthalpy $\{h_j\}$ at level j as a function of the enthalpy $\{h_{j-1}\}$ at level j-1. The values of $\{m_{j-1}\}$, $\{p_{j-1}\}$, and $\{w_{j-1}\}$ are assumed to apply at level j for the purpose of evaluating the fluid properties. In COBRA-II an equation derived from the combination of the axial and transverse momentum equations is discretized using a backward spatial differencing scheme, and the equation is solved directly for the crossflow distribution $\{w_j\}$ at level j as a function of the values of $\{m_{j-1}\}$, $\{w_{j-1}\}$, and $\{p_{j-1}\}$ at level j-1. The newly computed values of $\{w_j\}$ are compared with the old ones to determine if all the crossflows have converged to within an acceptable tolerance. If convergence does not occur, a flag is set up within the program to perform another iteration within the loop at the axial elevation of interest. Since the new value of $\{w_j\}$ is quite sensitive to the old

Fig. 4.1 Flow Chart for COBRA-II

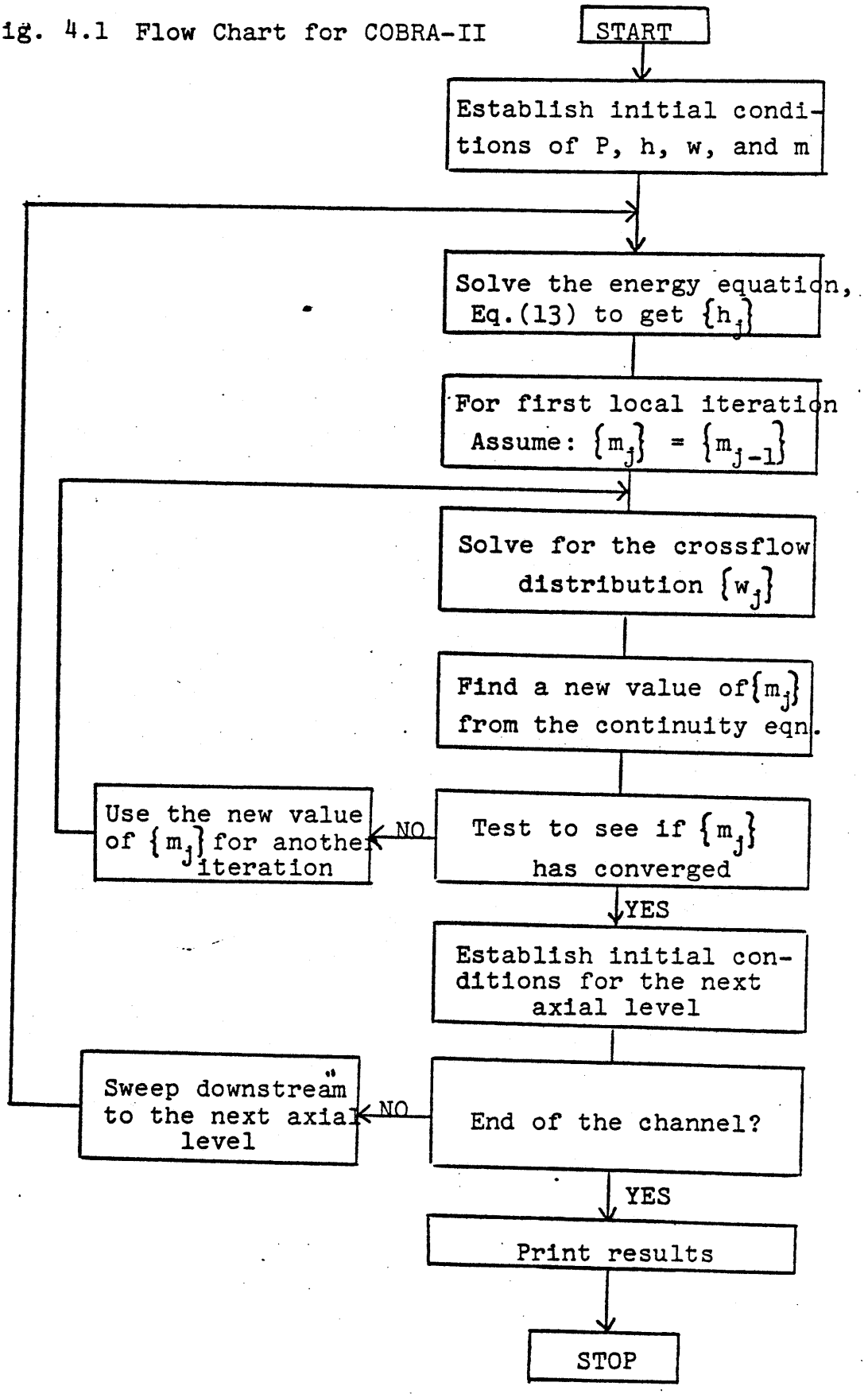
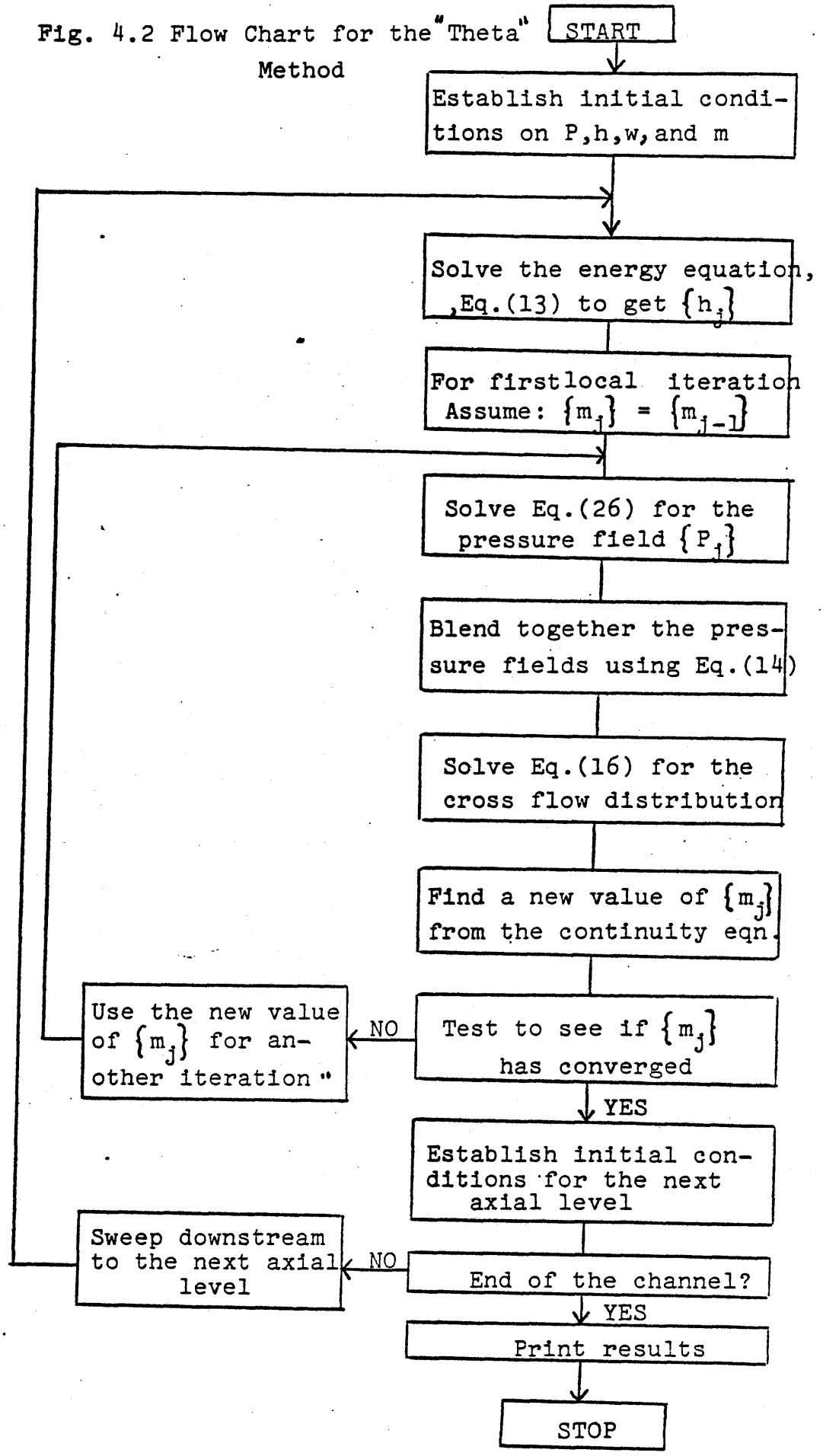


Fig. 4.2 Flow Chart for the "Theta" Method



one, $\{w_j\}$ is modified according to the algorithm

$$\{w_j\}^{(k)} = 1/2\ \{w_j\}^{(k-1)} + 1/2\ \{w_j\}^{(k)} \tag{75}$$

to presumably accelerate the convergence of the iteration, where k and k-1 are iteration indices for the crossflow at the same points in space and time. This value of the crossflow is then used to update the values of the axial mass flow rates via the continuity equation, and if convergence does not occur, the new values of the axial mass flow rates are used to update the coefficients in the difference equations to be solved for $\{w_j\}$. If convergence does occur, the pressure $\{p_j\}$ is computed from the axial momentum equation as a "back calculation" and the numerical scheme is allowed to sweep downstream to the next axial level.

Because of the fact that only the very simplest form of the transverse momentum equation

$$\{C|w|w\} = [s]\{p\} \tag{76}$$

is used in COBRA-II, the coefficient matrix of the difference equations used to compute the crossflow distribution is almost singular, and it is necessary to impose an upper limit on the space step size for a given value of the crossflow resistance coefficient [C] to prevent transverse flow loops. On the other hand, because of the fact that the values of $\{w_j\}$ are updated periodically to converge the local axial mass flow rates, it is also necessary for the largest eigenvalue of the coefficient matrix governing the iteration

on $\{w_j\}$ to have an absolute value of less than 1.0. This imposes another lower limit on the acceptable size of the axial mesh spacing for a given value of the crossflow resistance coefficient:

$$\Delta x > \frac{2[C]\{w_j\}}{[S][S]^T[u^*]} \tag{77}$$

where $[S][S]^T[u^*]$ is a matrix containing the axial velocity of the fluid carried by the diversion crossflow. If this second condition is not satisfied, errors grow in the flow solution and flow reversals may occur in an attempt to satisfy the momentum equations. Fortunately, this minimum value of Δx is much smaller than the values (i.e., 1 inch to 6 inches) that are usually desired for rod bundle calculations.

The theta method can be formulated to use essentially the same numerical procedure to govern the computation of the flow and enthalpy fields, except that the idea of the method is to construct a system of difference equations which can be solved directly for the pressure fields, so that the computation of the crossflow distribution is only a "back calculation." The restriction on the minimum size of the axial mesh spacing is avoided because the coefficient matrices governing the pressure field at each axial level of the core can be shown to be diagonally dominant for any value of the space step size. Reducing the time step size simply increases the diagonal dominance of the coefficient matrix and accelerates the rate of convergence of the

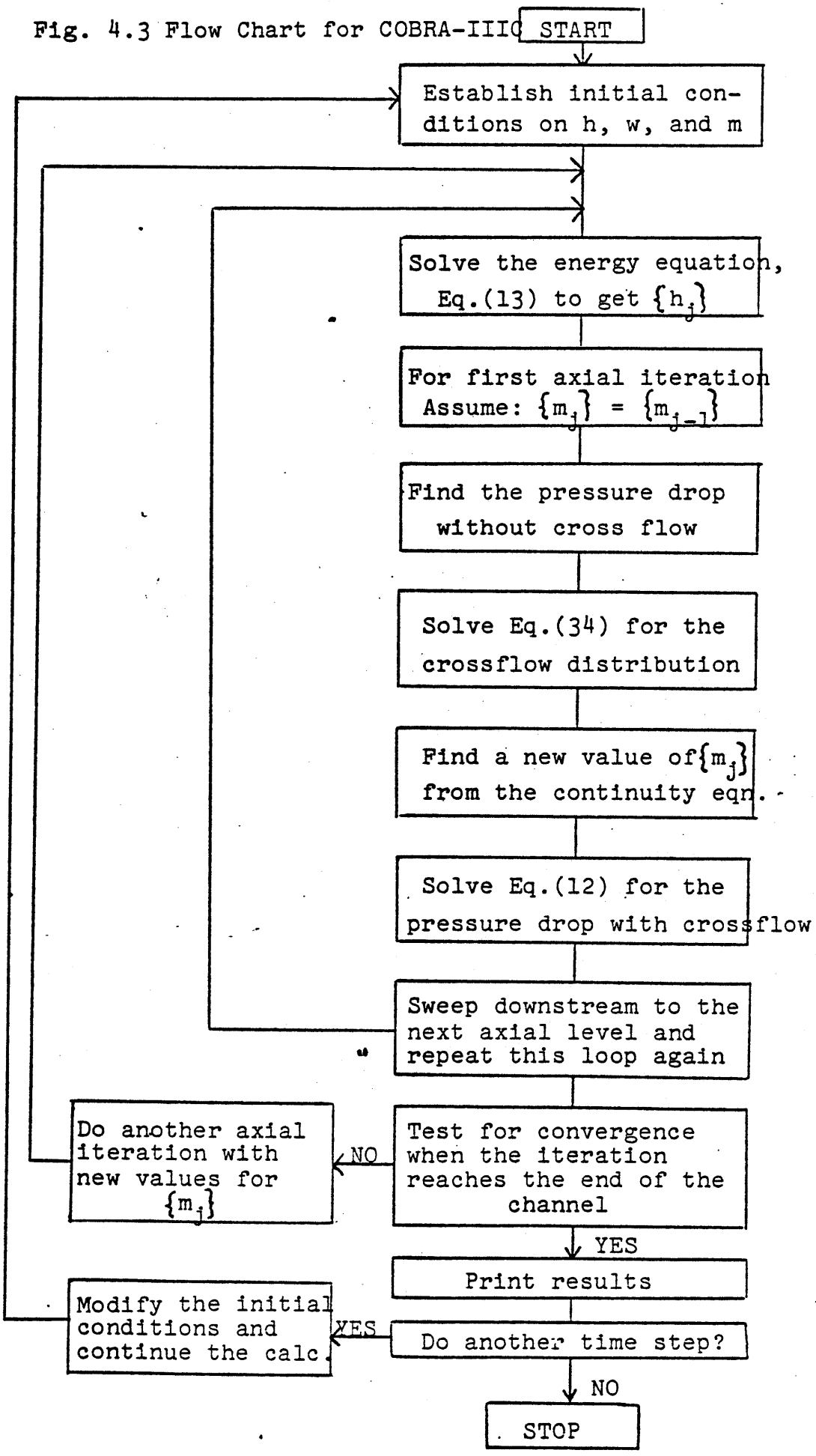
iteration. Consequently, the theta method allows a more complex system of conservation equations to be solved more efficiently without experiencing the major drawbacks of the initial value approach used in COBRA-II.

4.1.3 Multi-Pass Axial Iteration Schemes

For certain types of problems it is believed that a more consistent picture of physical reality can be obtained by constructing a "multipass" axial iteration scheme in which the flow and pressure fields are found by marching downstream several times in succession between the inlet and the outlet of the core and iterating on the flow fields until the change in the mass flow rates formed from successive axial iterations is less than some prescribed convergence criterion. This type of marching scheme is simply a generalization of a single pass locally iterative method in which an attempt is made to improve estimates of the flow and pressure fields on each successive sweep through the core. However, it is questionable in many cases whether this procedure is actually necessary because it involves updating the values of the variables at every point, although there may be only a few points (out of several thousand) where the iteration initially fails to converge. The primary motivation for using this type of axial iteration scheme is that it has the capability to propagate information from downstream locations to upstream locations

at the rate of one axial mesh point per iteration. Consequently, the effects of blockages and bulk boiling can be felt upstream of the points at which they actually occur if a backward differencing scheme, such as the one used in COBRA-IIIC, is applied to the pressure fields in the momentum equations. MAT method and the COBRA-IIIC method are similar to one another with respect to their use of boundary conditions on the system pressure and the computational procedures that can be employed to determine the flow and enthalpy distributions. Both methods may be based upon computing pressure differences rather than the absolute of the pressure, and a "boundary value" type of solution may be obtained with either method by updating the inlet pressure distribution on successive axial iterations through the core while the inlet mass flow rates and the system outlet pressure are assumed to remain constant. Thus, neither method requires an explicit knowledge of the inlet pressure distribution to begin a calculation. As shown in the flow chart in Fig. 4.3, a COBRA-IIIC computation is begun by sweeping from the inlet to the exit of each channel. With inlet information on the flow, crossflow, and enthalpy distributions given, the enthalpy can be advanced from axial j-1 to axial level j by solving the same explicit energy equation used by the MAT method, Eq. (13). For the first axial iteration the flow rate $\{m_j\}$ at level j is set equal to the flow rate $\{m_{j-1}\}$ from

Fig. 4.3 Flow Chart for COBRA-IIIC



level j-1; otherwise, the previous iterate is used. A system of equations for the crossflow distribution is generated by combining the axial momentum equation, Eq. (12), with the transverse momentum equation, Eq. (16), and the continuity equation, Eq. (11), to eliminate the lateral pressure difference $[S]\{p_{j-1}\}$ at level j-1. The lateral pressure difference $[S]\{p_j\}$ at the top of the node is then set equal to zero, and this system of equations is solved directly for the crossflow distribution $\{w_j\}$ using a first order implicit differencing scheme to approximate the spatial and temporal derivatives [7]. The fact that the lateral pressure differences at the top of each node are initially set equal to zero does not mean that the COBRA-IIIC code uses an iteration scheme to equalize these differences at the top of each plane of computational cells in the core, since the lateral pressure differences are found by a "back calculation" after the cross-flow is driven by an initial estimate, $[S]\{p_j\}-[S]\{p_{j-1}\}$, of the lateral pressure difference at each axial elevation. The axial mass flow rates $\{m_j\}$ at level j are updated using the continuity equation, Eq. (11), and the numerical scheme is allowed to sweep downstream to the next axial level, where this entire process is repeated again. When the calculation reaches the exit of each channel, a check is made to see if the flow distribution has converged at each mesh point to the tolerance which was originally specified.

If convergence has been achieved, the computation stops; otherwise, the entire computation is repeated again starting at the first axial level (e.g., the core inlet), and sweeping downstream to the last axial level (e.g., the core outlet). During the second and subsequent iterations, the most recent values of the axial mass flow rates are used in the equation for the computation of the crossflow distribution. This eliminates the uncertainties that were introduced into the solution by estimating the value of $\{m_j\}$ for the first iteration. However, none of the boundary conditions at the inlet of the core are changed because they are assumed to be known before the solution begins. Only the inlet pressure distribution(which is not a boundary condition) is "floated" to adjust for changes in the total pressure drop through the core that are created by successively better estimates of the crossflow distribution. The number of axial iterations required for this procedure to converge depends upon the type of problem being solved, and is obviously also a function of the flow convergence criterion that is used, since the distance the pressure difference $[S]\{p\}$ is propagated upstream depends upon the amount the axial mass flow distribution is perturbed by the presence of grids, blockages, or the incidence of boiling at downstream locations.

The numerical procedure that must be implemented to allow the MAT method to be used with a multi-pass axial

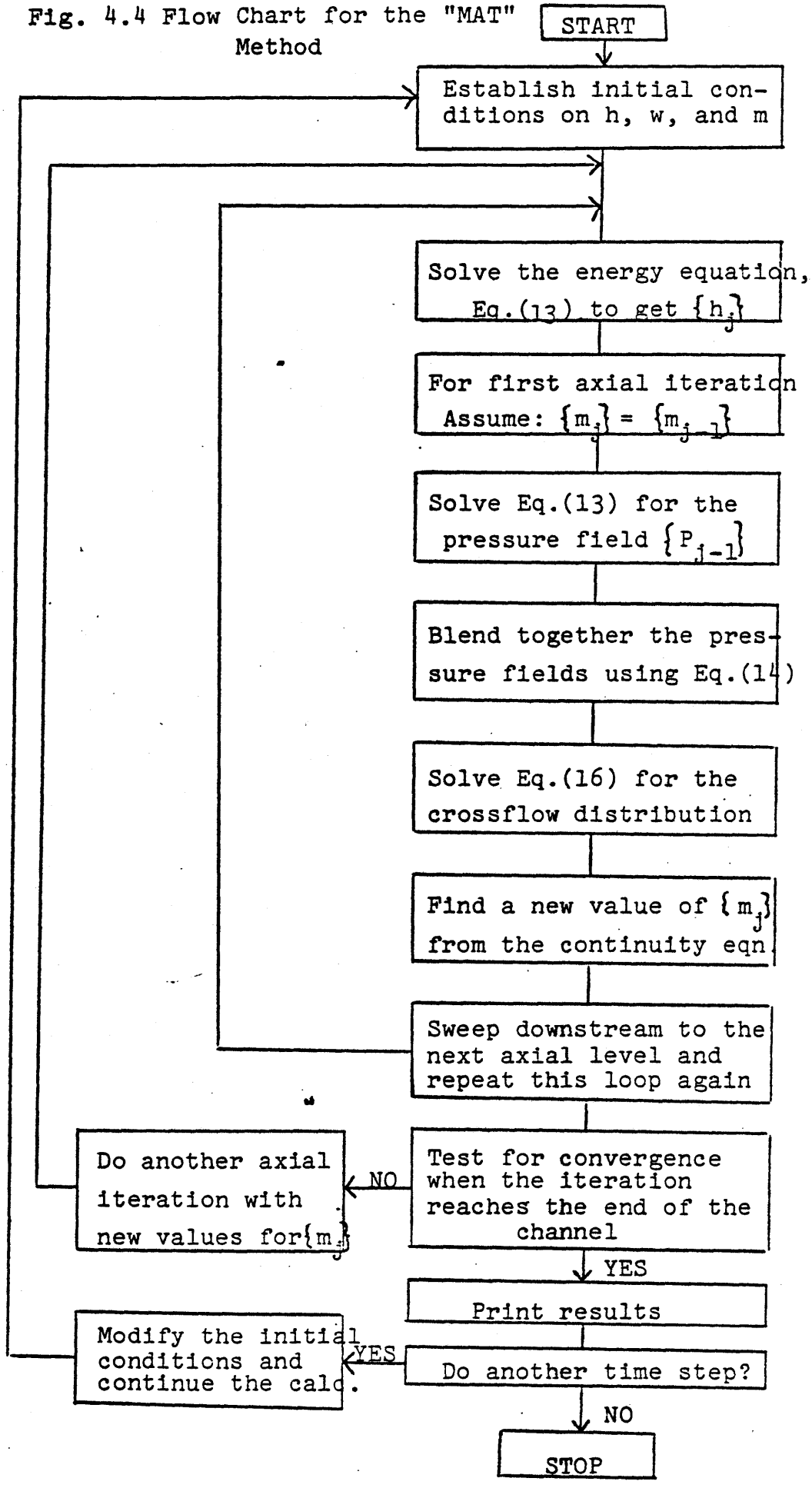
iteration scheme is essentially the same as the procedure described above. As shown in the flow chart in Fig. 4.4, the energy equation, Eq. (13), is solved explicitly for the enthalpy $\{h_j\}$ at axial level j using the value of the enthalpy $\{h_{j-1}\}$ already computed at axial level j-1. The momentum equations, Eqns. (12) and (16), are then combined together with the continuity equation,(Eq. 11), to generate a system of equations Eq. (67) for the pressure distribution $\{p_j\}$ as a function of the axial mass flow rate $\{m_j\}$. The values of $\{m_j\}$ in both the source vector and the coefficient matrix of these equations are unknown, but they can be initially estimated and updated using the iterative process described previously. To perform an axial iteration, the conservation equations used by the MAT method are solved at each axial level as the solution scheme sweeps downstream between the inlet and the outlet of each channel. As in the original version of the theta method, the pressure distribution $\{p_{j-1}\}$ found by solving Eq. (67) is blended together with the pressure field $\{p_j\}$ from the top of each axial level to form the composite pressure field given by Eq. (14):

$$\{p\} = \theta\{p_j\}+(1-\theta)\{p_{j-1}\} \tag{78}$$

The crossflow distribution at each axial elevation is then found from the transverse momentum equation by rewriting it as

$$\{w_j\} = [D_j]^{-1}\{O_j\} + \left(\frac{s}{\ell}\right)[D_j]^{-1}[S]\left\{\theta\{p_j\} + (1-\theta)\{p_{j-1}\}\right\} \tag{79}$$

Fig. 4.4 Flow Chart for the "MAT" Method



where $[D_j]^{-1}$ is the diagonal matrix defined by Eq. (64) and $\{O_j\}$ is a vector containing the other forcing terms (see Eq. (65)). As indicated previously, the additional step requires very little computational effort because the inverse of $[D_j]$ can be found by inspection. Thus the pressure distribution is found directly at each step in the iteration while the computation of the crossflow distribution is a "back calculation." As in COBRA-IIIC, the value of $\{p_j\}$ (or equivalently, the value of the lateral pressure difference $[S]\{p_j\}$) in Eqns. (66) and (79) is not initially known, but it can be set equal to a constant (such as zero to avoid the necessity of computing the source term $[I-\vartheta M']\{p_j\}$) for the first axial iteration, and then updated so that its new value can be used for successive iterations. Because of the fact that the MAT method applies a backward differencing scheme to the momentum equations, downstream effects can be propagated upstream at the rate of one axial mesh point per iteration.

4.2 Considerations for Transient Situations

In both the COBRA-IIIC approach and the MAT methodology, steady state calculations are performed by setting the time step Δt equal to some arbitrarily large value, since the difference equations used by both methods are temporally implicit and inherently stable for large time steps. Transient calculations are performed in the same way but

for a selected time step size Δt. "Boundary conditions" and other forcing functions are set to their desired values at the beginning of each new time step; then, the calculation sweeps through the core for the number of axial iterations needed to achieve a convergent flow distribution. The converged solution is used for the new initial condition and the same procedure is continued for all time steps until the end of the transient is reached.

CHAPTER V

5.1 The COBRA-IIIP/MIT Code

The previous chapters of this thesis have been concerned primarily with discussing the mathematical formalism upon which the numerical methods that have been developed are based. The purpose of this chapter is to compare the computational efficiency of these methods to one another and to outline the way in which these methods have been integrated into the computational logic of a computer code called COBRA-IIIP/MIT. As discussed in detail in an earlier publication [16], this code is based upon the same conservation equations and physical models that are used in COBRA-IIIC, but due to improvements in the method of coding, and the use of the numerical methods that have been developed in this thesis, it will be shown that it is now possible for the code to solve much larger problems than previous codes with much greater speed.

The first version of the COBRA-IIIP/MIT code was based upon the theta method outlined in Chapter II. This version of the code was suitable for modeling operational reactor conditions where the crossflow was less than 5 or 10% of the axial mass flow rate, but was not designed or was ever intended to be used to analyze the effects of severe flow blockages. This was primarily because the $\{a_{j-1}'\} = \{a_j'\}$ approximation outlined in section 3.1.1 could only be

justified a priori for cased in which the "higher order" $\{2m_{j-1}\Delta m\}$ and $\{\Delta m^2\}$ terms in the axial momentum equation could be ignored. When problems are encountered where the crossflow becomes nearly as large (or larger) than the axial mass flow rate, the $\{a_{j-1}'\} = \{a_j'\}$ assumption is presumably invalidated, and better approximations to the flow-squared terms, such as those used by the MAT method, have to be made. Consequently, a more general version of the COBRA-IIIP/MIT code was set up and in this version of the code, a computational algorithm was developed in which the conservation equations were solved by the MAT method rather than by the theta method.

5.2 Rational for the Axial Iteration Scheme used by the COBRA-IIIP/MIT Code

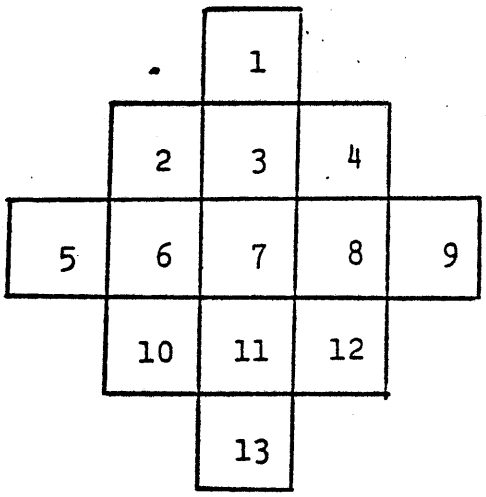
Although the MAT method has been shown to be compatible from a numerical point of view with multipass axial iteration schemes in which information is propagated from downstream locations to upstream locations at the rate of one node per iteration, this type of axial iteration scheme is not contained in the latest version of the code. This is because it is believed to be computationally wasteful to iterate over the entire axial height of the core many times in succession if there are only one or two points (out of several thousand in some cases) where the iteration initially fails to converge. Further, it is believed that

this computational procedure must be avoided if future improvements are to be made to the MEKIN code [17] especially since the use of a thermal hydraulic method with a multipass axial iteration scheme can easily cause the running time of the code to become prohibitive if it is applied to problems where a large number of axial iterations have to be performed during each sweep between the neutronics and the thermal-hydraulics[18]. Consequently, to maintain a reasonable balance between computational efficiency and physical accuracy, the MAT method has been implemented into COBRA-IIIP/MIT as a single pass, locally iterative method. Thus additional iterations are only performed at those planes in the core where they are needed. This type of axial iteration scheme coupled to the inherent computational efficiency of the MAT method has enabled the code to solve problems having as many as 625 channels with arbitrary geometrical shapes, and with no limit to the number of time steps or space steps that can be used. However, because of the fact that the backward differencing scheme discussed in section 4.1.3 has been retained in this application of the MAT method, the new version of the COBRA-IIIP/MIT code does not require an explicit knowledge of the inlet pressure distribution to obtain a meaningful solution to the entire system of conservation equations. Thus the boundary conditions required to run the code are now exactly the same as those used by COBRA-IIIC.

5.3 Solution Schemes

5.3.1 Iterative Methods - Successive Overrelaxation

Since the stucture of the coefficient matrix which governs the pressure field in each of the methods that has been developed is a strong function of the geometrical arrangement of the cells in the lattice, the types of numbering schemes that can be applied to these cells can have an important effect upon the overall computational efficiency of the methods. For example, in cases where the cells in the lattice are arranged in rectangular arrays, both of these coefficient matrices will be diagonally dominant, positive definite, irreducible matrices if the cells are numbered consistently (i.e., either row by row or column by column as shown in Fig. 5.1) and if each cell is connected to no more than four other neighboring cells. In this case, the entries in each matrix can be ignored if they have a value of zero, and the remaining entries with non-zero values can be compressed into a 5-striped array in the manner shown in Figure 5.2. The pressure distribution may then be found by applying the method of successive overrelaxation to the non-zero entries within the array. The successive over-relaxation in COBRA-IIIP/MIT is carried out on a pointwise basis to account for the fact that the cells may have different sizes and geometrical



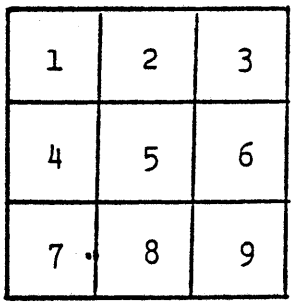


Fig. 5.1 Consistent Cell Numbering Schemes

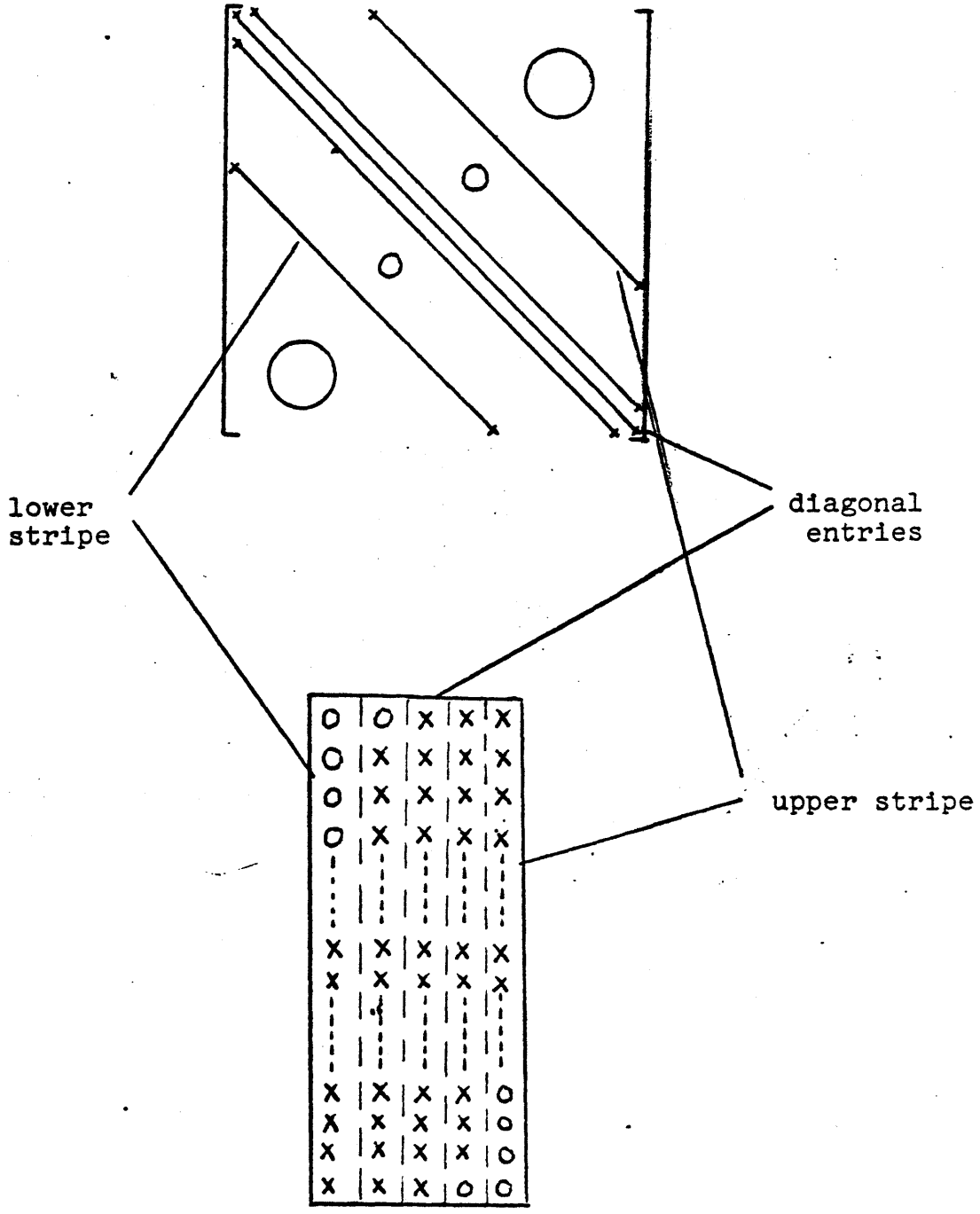


Fig. 5.2 Five-Striped arrays used for
successive overrelaxation

arrangements. However, it is recognized that it may be more desirable for rectangular geometries to carry out the relaxation line by line. The relaxation factor governing this iterative scheme may be supplied by the user as part of the input data required to execute the code, or it may be computed by the computational subroutines built into the code itself. The optimum relaxation factor is found by monitoring the behavior of the residuals formed by taking the difference between successive iterates of the pressure field, and applying the method of Carré [19] to infer the value of the relaxation factor which minimizes the spectral radius of the iteration matrix.

5.3.1.1 Successive Overrelaxation for Assembly-Sized Nodes

It has been found that the optimum relaxation factors for both the theta method and the MAT method are a strong function of the ratio of the axial mesh spacing (Δx) to the cross-sectional flow area"[A_j] of the cells. For an axial mesh spacing of six inches and a flow area representative of an entire PWR fuel assembly (i.e., $\approx$40 square inches) the optimum relaxation factor is usually less than 1.1, which indicates that the spectral radius of the iteration matrix is close to 0.0, and that errors introduced into the iteration

are damped out very quickly [20]. For problems of this type, the transverse pressure distribution usually converges to three decimal places in less than eight radial iterations, regardless of the shape of the radial or axial power distributions and the number of rod spacers (i.e., grids) oriented perpendicular to the direction of the flow.

Table 5.1 shows the number of "inner iterations" needed to converge the pressure field to three decimal places as a function of axial position for a reactor core 120 inches high in which only one computational cell is used to represent each fuel assembly in the transverse plane. The core has five different types of grids and 128 fuel assemblies. An axial mesh spacing of six inches is used for this case. The rapid convergence of the pressure field shown here is indicative of the convergence rates of both methods when they are applied to problems in which it is desirable to represent large numbers of fuel assemblies explicitly.

5.3.1.2 Successive Overrelaxation for Subchannel-Sized Nodes

It is also necessary in reactor analysis to consider cases in which the method of subchannel analysis is used to represent each subchannel by an individual computational cell. Several cells that may be used for this purpose are shown in Fig. 5.3. For problems of this type, the ratio of the axial mesh spacing tothe cross-sectional flow area is much larger for a given value of Δx than when one cell is used to represent an entire fuel assembly. As equations (26),

Table 5.1. Number of Iterations Needed to Solve Problems with Assembly-Sized Cells

Node Number	Axial Elevation (inches)	Number of Iterations
1	6	6
2	12	6
3	18	7
4	24	6
5	30	8
6	36	7
7	42	6
8	48	6
9	54	6
10	60	6
11	66	7
12	72	6
13	78	6
14	84	8
15	90	7
16	96	6
17	102	6
18	108	6
19	114	6
20	120	8

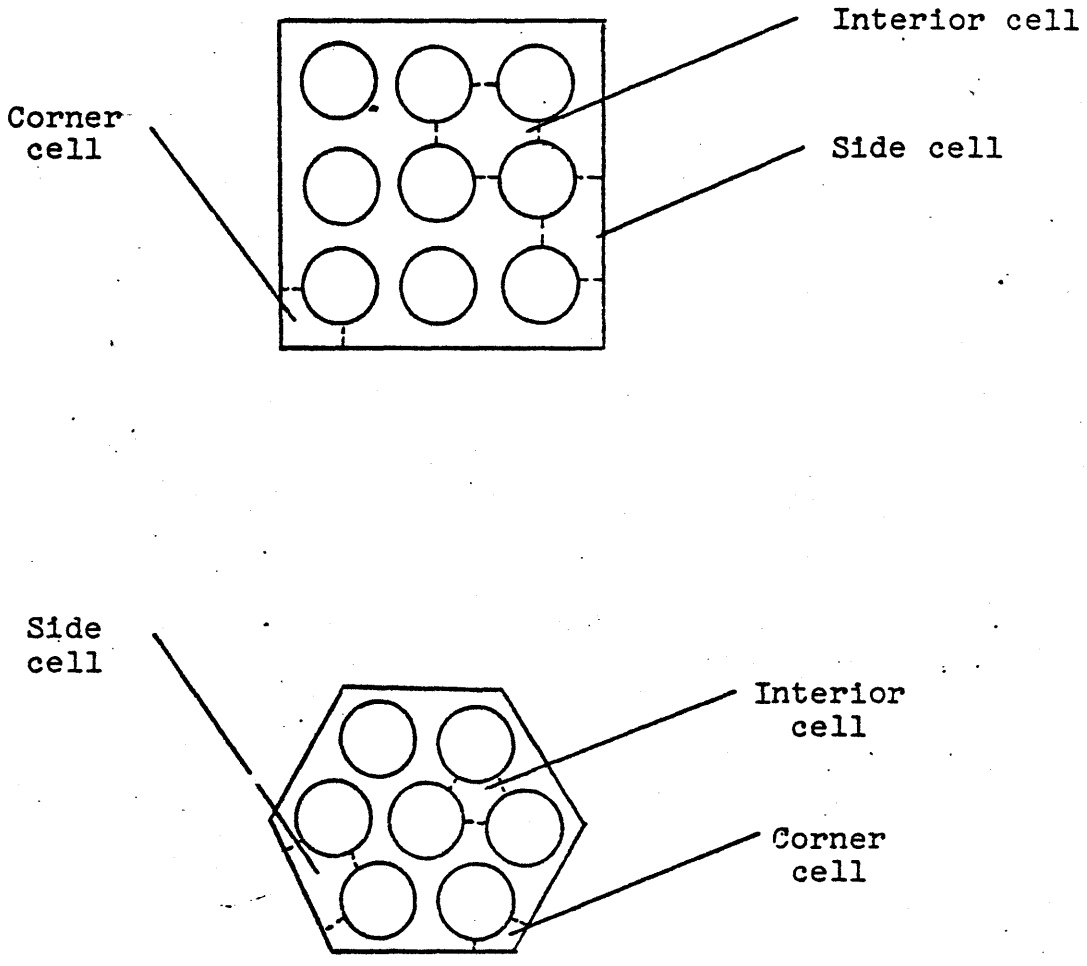


Fig. 5.3 Cells for the method of subchannel analysis

(27), (67), and (68) show, the large value of this ratio tends to make the sum of the off-diagonal terms in the coefficient matrices almost comparable in magnitude to the diagonal terms. In this case, the diagonal dominance of the coefficient matrix is reduced and the spectral radius of the matrix governing the iterationon the pressure field is much closer to 1.0. For a typical problem of this type, a PWR subchannel may have a cross-sectional flow area of a few tenths of a square inch, and the axial mesh spacing for a fuel management type of analysis is usually chosen to be about six inches. Using the method of Carré, it has been found that this type of situation results in an optimum relaxation factor between 1.6 and 1.8. From a practical point of view, the fact that the relaxation factor is so large means that a great many more iterations are required to converge the pressure field to an acceptable tolerance if an iterative type of inversion scheme is used. Table 5.2 shows the number of "inner iterations" required by both the MAT method and the original theta method to converge the pressure field to three decimal places for the small section of the PWR fuel assembly shown in Fig. 5.4. In both cases, the number of iterations performed on the pressure field appears to be affected by the presence of the grids. It should be noted that the MAT method requies somewhat more iterations for the given input parameters, but as indicated previously, the MAT method is considerably more stable and

Table 5.2. Number of Iterations Needed to Solve Problems with Subchannel-Sized Cells

Node Number	Axial Elevation (inches)	Grid Position X = Yes	Number of Iterations for MAT	Number of Iterations for Theta
1	6	X	49	43
2	12		37	40
3	18		9	10
4	24		7	8
5	30		7	7
6	36		6	6
7	42	X	56	49
8	48		50	46
9	54		7	7
10	60		6	6
11	66		6	6
12	72	X	52	51
13	78		49	44
14	84		6	7
15	90		6	6
16	96	X	41	39
17	102		40	38
18	108		7	8
19	114		6	6
20	120	X	60	52

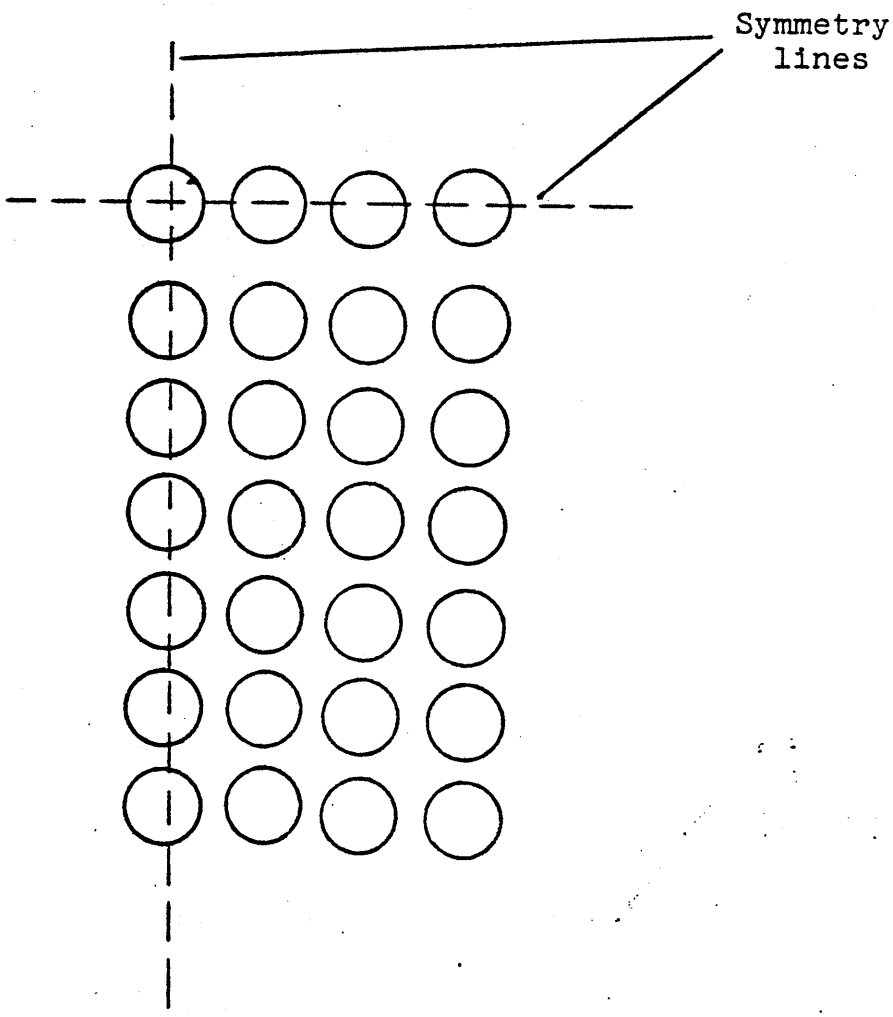


Fig. 5.4 Small section of a PWR fuel assembly

does not require detailed information about the boundary conditions on the pressure distribution at the core inlet to determine the flow and enthalpy fields.

5.3.2 Direct Solution Schemes-Gaussian Elimination

Whereas the method of successive overrelaxation can be used very effectively to determine the pressure fields in a lattice where the cells are arranged in periodic, regular arrays, it has the drawback that it is difficult to apply to problems such as the one shown in Fig. 5.5 where the cells are lumped together into clusters having a variety of different sizes and geometrical shapes. This is due to the fact that the coefficient matrices governing the pressure fields do not have a simple or predictable bard structure, so that it is necessary to operate on many of the zeros in each matrix* to ensure that each cell interacts with all of its neighboring cells in the lattice. Moreover, it is frequently necessary to consider a reactor lattice in which the rods are packed very tightly together, so that the subchannels are considerably smaller than those encountered in the analysis of a pressurized water reactor. In this case, the spectral radius of the iteration matrix is very close to 1.0, and the iterative solution scheme takes many iterations to converge. For problems of this type, it may be more efficient from a computational point of view to abandon the method of successive

---

* but not as many as COBRA-IIIC does

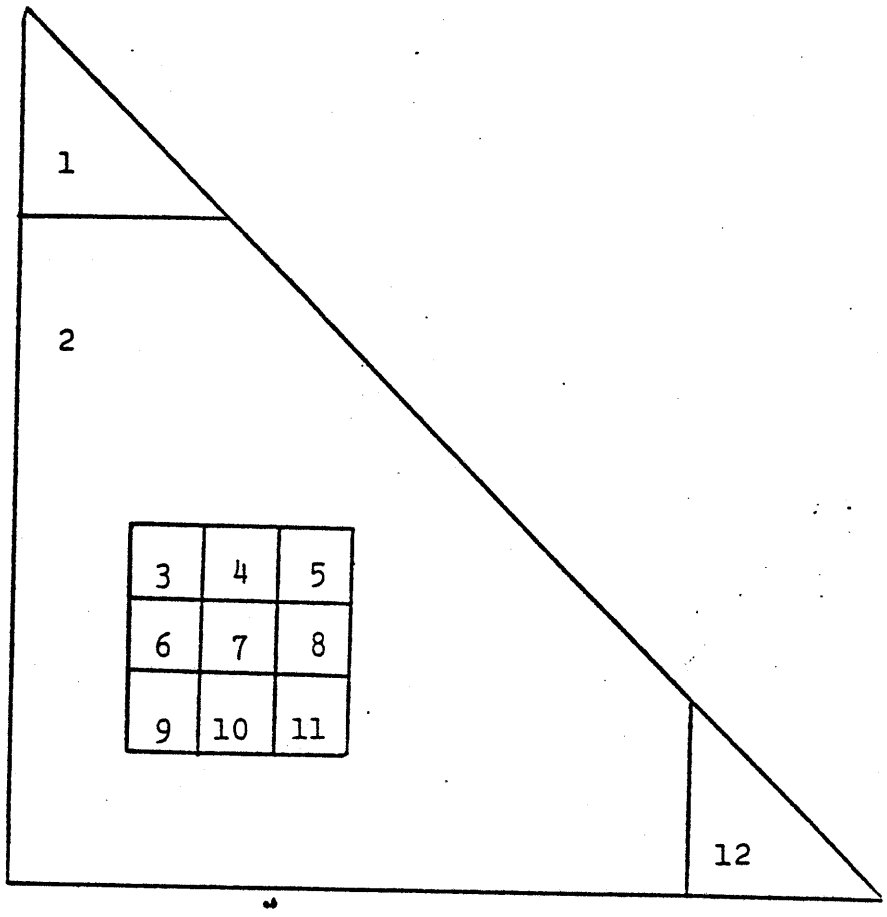


Fig. 5.5 Cells for a mixed lattice calculation

overrelaxation and to solve Eq. (26) to Eq. (67) by the method of Gaussian elimination. Since the Gaussian elimination is simply a procedure for manipulating the coefficients of the difference equations to give a coefficient matrix which is lower triangular, the number of operations required to reduce the coefficient matrix to this form is a very sensitive function of the type of cell numbering scheme that is used. For example, in some numbering schemes such as the one shown in Fig. 5.6, the cells may be connected to one another in such a manner that the coefficient matrix has no preceptible band structure. In this case, it is necessary to operate on virtually all the entries in the matrix (including all the zeros) to reduce it to lower triangular form. This is extremely inefficient from a computational point of view because the number of operations required to solve for the pressure field can be shown to be proportional to the number of channels cubed [20]. Fortunately, it is possible to devise cell numbering schemes such as the one shown in Fig. 5.7 which tend to compress the non-zero coefficients governing the interactions between adjacent channels into a much narrower stripe centered around the diagonal so that the coefficients above and below the stripe have entries that are exclusively zero. An example of a coefficient matrix of this type is illustrated in Fig. 5.8(a). In this case, the width of the band (MS) is determined by the code from simple geometrical arguments, and the computational

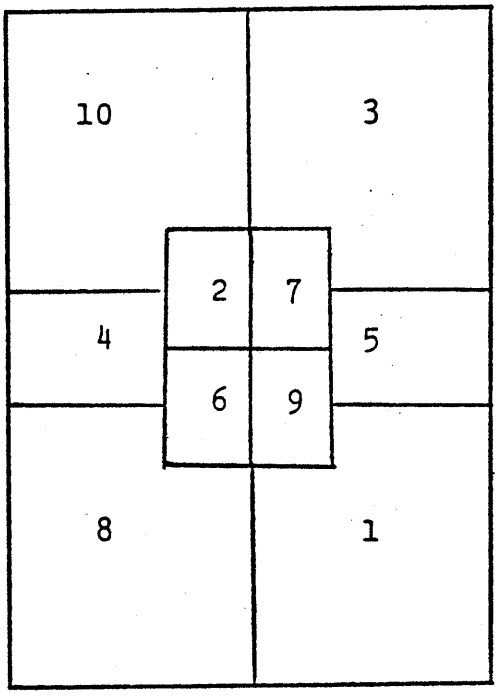


Fig. 5.6 An arbitrary cell numbering scheme

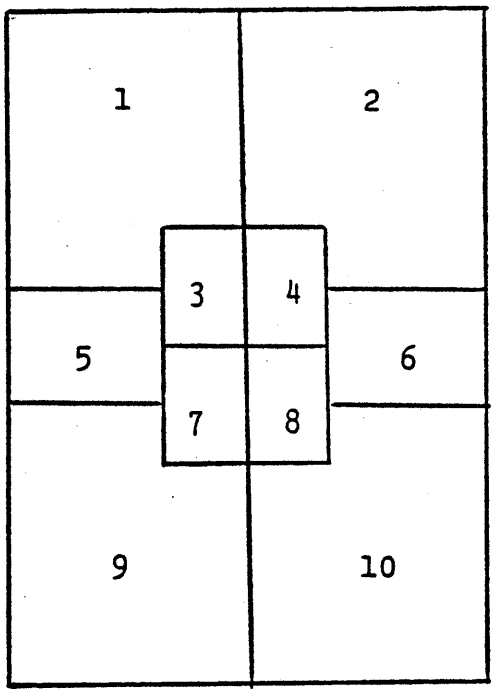


Fig. 5.7 An optimum cell numbering scheme

subroutines within the code only generate the coefficients inside the band that have non-zero values. The band is then stored by the code as an N by MS array (see Fig. 5.8(b)) and the Gaussian elimination is performed by operating upon only the entries contained within this array. Since the number of operations required to set up the coefficient matrix is proportional to the number of channels, the number of operations required to decompose the matrix is proportional to the number of channels times the square of the band width, and the number of operations required to solve the remaining system of matrix equations for the pressure field is proportional to the number of channels squared, the total central processing unit (CPU) time needed to complete this entire process can be written as

$$\text{CPU time} = C_1 N + C_2 N(MS)^2 + C_3 N^2$$

where N is the number of channels and C_1, C_2 and C_3 are constants that are problem dependent. For most problems empirical studies indicate that the values of C_1, C_2 and C_3 result in a computation time that is proportional to the number of channels to the 1.6 power. Thus it can be seen that a considerable reduction in the running times of both the theta method and the MAT method can be achieved by simply taking advantage of the sparsity of the coefficient matrix and by adopting a cell numbering scheme in which the non-zero coefficients within each matrix are compressed into a narrow band centered around the diagonal.

Fig. 5.8(a) Coefficient matrix used for Gauss elimination

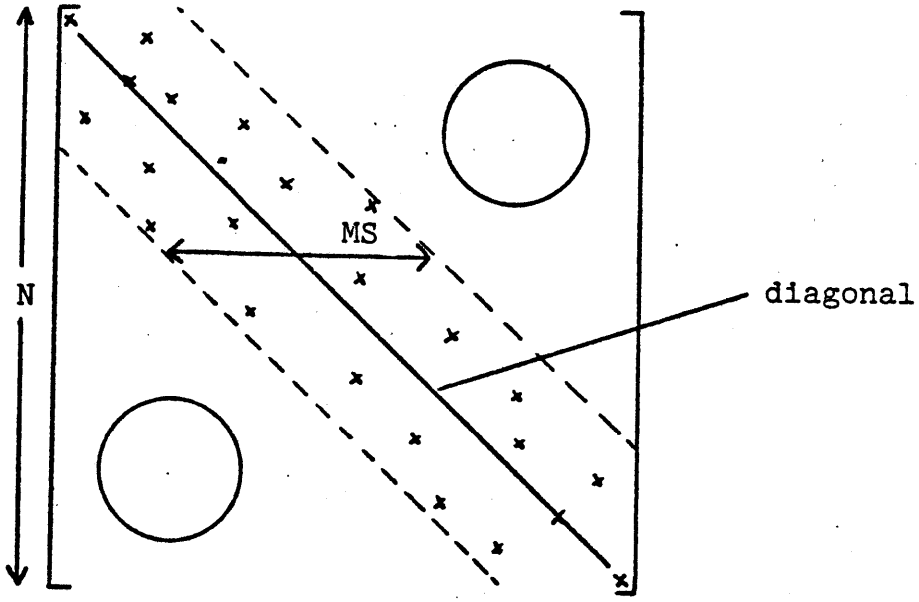
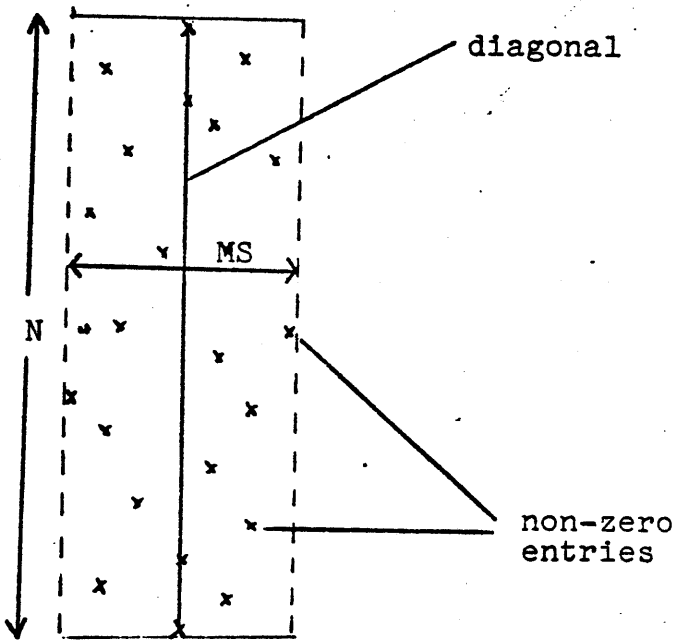


Fig. 5.8(b) Compressed coefficient matrix used for Gauss elimination



5.4 Execution Time Comparisons

Table 5.3 gives a comparison of the computation time needed by various codes and methods in use today to compute the crossflow distribution as a function of the number of computational cells. These results indicate that the versions of the COBRA-IIIP/MIT code that have been based upon either the MAT method or the theta method are considerably faster and more efficient than their predecessors. Whereas the crossflow computation time of the COBRA-IIIC code increases as the number of cells cubed, and the crossflow computation time of COBRA-IIIC/MIT still increases as the 1.6 power of the number of cells, the computation time required to compute the crossflow distribution with COBRA-IIIP/MIT is approximately a linear function of the number of cells when an iterative solution scheme is used. If the equations for the pressure field are solved by Gaussian elimination, rather than successive overrelaxation, the direct inversion scheme tends to make the computation time proportional to the number of channels to the 1.6 power, but as stated previously, the total CPU time is still considerably less than that of COBRA-IIIC/MIT because the band width of the coefficient matrix governing the pressure field (section 2.6) is only half that of the COBRA-IIIC/MIT crossflow coefficient matrix for problems with a large number of computational cells. It is also important to realize that the execution times given in rows 1, 2, and 5 are independent

Table 5.3. Crossflow Solution Times for Various Thermal Analysis Codes

Code	Method of Solution	Number of Channels					
		16	64	128	200	300	400
COBRA-IIIC	Gauss Elimination with no compression of crossflow coefficient matrix	1.7	128.49*	987.36*	3293.61*	10250*	31980*
COBRA-IIIC/MIT	Gauss Elimination with full compression of crossflow coefficient matrix	.17	1.38	9.09	19.67	39.80*	81.24*
COBRA-IIIP/MIT with MAT Method	Successive overrelaxation with optimized relaxation factor	.15	.44	1.23	1.82	2.65*	3.62*
COBRA-IIIP/MIT with Theta Method	Successive overrelaxation with optimized relaxation factor	.15	.42	1.28	1.78	2.67*	3.59*
COBRA-IIIP/MIT with MAT method or theta method	Gauss Elimination with full compression of pressure coefficient matrix	.17	.76	2.16	4.14	7.95*	16.53*

Notes: All results are for 10 axial levels on an IBM 370/165 with the H compiler. Note that the results of these timing runs may vary by ± 10% during the course of a day due to changes in the work load on the system.

* For economic reasons these results are estimated by extrapolation.

of the axial mesh spacing, the size of the cells, and the number of rod spacers (i.e., grids) because they are based on the use of Gaussian elimination, whereas the execution times for successive overrelaxation can be somewhat longer or shorter than those given here if the size of the cells is changed or if one fails to obtain a good estimate for the optimum relaxation factor before proceeding with the iterative solution scheme. Hence the crossflow solution time of the COBRA-IIIP/MIT code will always lie somewhere between the times given in rows 3 and 5 , except in cases where the method of successive overrelaxation converges more rapidly. Finally, it can be seen that there is no significant difference between the execution times of COBRA-IIIP/MIT when the MAT method and the theta method are used, in spite of the fact that the MAT method is considerably more general and more stable. Users who wish to make timing runs with the codes may do so using the simple timing subroutine shown in Table 5.4.

5.5 Effect of the Axial Iteration Scheme on the Solution Time

For the sake of consistency, the results in Table 5.3 have been presented for cases in which all the codes have been allowed to perform only one axial iteration. In practice, it is often found that several iterations may be needed to converge the methods that have been discussed in this thesis to a self-consistent flow distribution. However, the amount

Table 5.4. A Simple Subroutine for Timing Runs

```
                SUBROUTINE PRNTIM (IN)

C.  This subroutine may be used to determine the crossflow
C.  solution time and the total execution time of the
C.  COBRA-IIIP(MIT) code by calling it within the code at
C.  selected points of interest



      CALL TIMING(IT)
      IF(IN.EQ.0) GO TO 100
      CPUTIM=FLOAT(IT-ITT)*0.01
      WRITE(6,1000) IN,CPUTIM
1000  FORMAT(1X,'*** PRNTIM ***   (',I1,')',F7.2,' SEC.',/)
      RETURN
      END
```

of CPU time required by COBRA-IIIP/MIT to achieve this convergence is often considerably less than COBRA-IIIC or COBRA-IIIC/MIT due to the use of the single pass, locally iterative axial iteration scheme contained within the code. This scheme has usually been found to reduce the computation time by an additional factor of 3 or 4 compared to the other codes, depending upon the type of problem being solved. Finally, it has been found that the total execution time of COBRA-IIIP/MIT is approximately 70% or 80% greater than the crossflow solution times shown in Table 5.3. This appears to indicate that the new code and the methods upon which it is based are considerably faster and more efficient than their predecessors.

CHAPTER VI

6.1 Concepts of Convergence and Stability Applied to Computational Fluid Mechanics

In many other branches of nuclear engineering, such as those which are concerned with solving problems in reactor physics, structural mechanics, and heat transfer, the differential equations governing the behavior of physical systems are well known and well understood. Consequently, a considerable amount of time and effort has been devoted to developing analytical techniques for predicting the stability, the convergence, and the consistency of numerical methods and differencing schemes for solving these conservation equations. The purpose of this chapter is to summarize the attempts that have been made to extend these techniques into the field of computational fluid mechanics, and to show in many cases that it is virtually impossible to use the same techniques to prove the convergence or the stability of the numerical methods that have been developed to solve the conservation equations used by the COBRA-IIIC code.

6.2 Analytical Proofs of Stability

Although a great deal of work has been done in recent years to attempt to prove the stability of numerical schemes for solving the fluid conservation equations, a careful study of the literature shows that this work has been

restricted almost exclusively to a Von Neumann stability analysis of the differential forms of the conservation equations used by primitive variable techniques such as "MAC" [12] and the stream function-vorticity approach [21] for the case of incompressible flow (ρ = constant). This approach has recently been extended by Porsching [22] to deal with a primitive variable technique for the case of a thermally expandable fluid, but no general stability analysis has ever been applied successfully (as far as it is known) to the forms of the conservation equations that are used by the COBRA-IIIC code. Rowe [8] considered the stability of the initial value approach used by COBRA-II and was able to show that the iteration governing the crossflow distribution at each axial level of the core could be made to be stable for a steady state case in which the spatial and temporal acceleration terms in the transverse momentum were ignored. Hansen and Khan [23] then attempted to extend Rowe's work to time dependent cases by considering a more general time dependent form of the conservation equations coupled to the multi-pass axial iteration scheme used in COBRA-IIIC. They failed to prove the stability of the method using a Von Neumann stability analysis because of the complex, non-linear form of the equations and ultimately recommended that the stability of the code should be determined by performing an unspecified set of numerical tests. While

the author of this thesis has been somewhat more successful in showing that both the crossflow distribution and the transverse pressure distribution can be found using iterative solution techniques because the coefficient matrices governing these iterative schemes have diagonal dominance, it is not believed to be possible to show (except by running the code, of course) that the overall axial iteration should also converge. Consequently, it is felt that the stability of the numerical methods developed in this thesis must be determined by numerical tests rather than by analytical means.

6.3 Numerical Tests of Stability

To explore the stability of these methods a very comprehensive set of numerical experiments have been performed. In these experiments the behavior of problems having as many as 600 channels has been studied, and it has been found that both the theta method and the MAT method are stable for all spatial differencing schemes in which the crossflow distribution is driven by a pressure field which is at least as implicit as the pressure field obtained by taking the average of the pressure distributions that exist at the top and the bottom of each plane of computational cells. From Eq. (26) it can be seen that this condition is satisfied for the theta method when it is formulated as an initial value method for values of θ greater than or equal to 0.5. Conversely, the same condition

is satisfied for the MAT method and for the theta method when it is formulated as a boundary value method for values of θ less than or equal to 0.5. More implicit spatial differencing schemes have always been found to be stable.

As far as the temporal differencing schemes are concerned, it appears that both methods are unconditionally stable since the difference equations governing the temporal behavior of the system are fully implicit.

However, from a practical point of view, it has been found that the MAT method is apparently able to handle much more severe cases than any version of the theta method before it becomes unstable. For example, MAT has successfully considered problems in which flow blockage simulations have created axial mass flux differences as great as 1000 to 1 between adjacent computational cells in the core, whereas the theta method, when it is formulated as an initial value approach, appears to become unstable for axial mass flux differences as low as 5 or 10 to 1. Thus it is believed that the MAT method should be used to analyze a problem that contains a severe flow blockage, and that it can be applied equally well to analyze problems that involve only small departures from operational reactor conditions. A summary of the stability requirements for the methods that have been discussed here is given in Table 6.1.

Table 6.1. Stability Requirements for Several Types of Solution Techniques

Method and Type of Solution Scheme	Spatial Stability Condition	Temporal Stability Condition
Theta Method -Initial Value Approach-	$\theta \geq 0.5$	None Method is Fully Implicit
Theta Method -Boundary Value Approach-	$\theta \leq 0.5$	None Method is Fully Implicit
MAT Method -Initial Value Approach-	$\theta \geq 0.5$	None Method is Fully Implicit
MAT Method -Boundary Value Approach-	$\theta \leq 0.5$	None Method is Fully Implicit
COBRA-II -Initial Value Approach-	See Eq.(77) for condition on minimum Δx Maximum $\Delta x \leq 1$ or 2 feet	Not Applicable Method is not Time Dependent
COBRA-IIIC -Boundary Value Approach-	None Method is Fully Implicit	None Method is Fully Implicit
COBRA-IV-I -"MAC" Methodology-	None-COBRA-IIIC Approach used for Steady State Conditions	Courant condition Method is Fully Explicit

6.4 Analytical Proofs of Convergence

Although a great deal of time and effort has been spent in recent years to develop numerical methods for solving the set of fluid conservation equations that characterizes the steady state and transient thermal hydraulic performance of fuel pin bundles and reactor cores, very little progress has been made to determine if these methods actually converge to unique and physically realistic solutions of the conservation equations. Proofs of convergence in computational fluid mechanics have been restricted almost exclusively to linearized versions of the Navier-Stokes equations for the case of incompressible flow [24]. These proofs have consisted of applying a linear Von Neumann stability analysis to the numerical methods that have been developed to show that they are stable, and then argueing that if the difference approximations are a consistent approximation to the differential equations as the mesh spacing approaches zero, then they must satisfy the Lax theorem [25]. and converge to the true solution of the differential equations. The major drawback of this approach is that it has not been extended to the true non-linear form of the differential equations and does not appear to be capable of proving the convergence of methods for the more general case of compressible flow [21] which is, of course, considered in the COBRA-IIIC code by means of the equation of state,(Eq. 15).

6.5 Numerical Tests of Convergence

The methods that have been discussed in this thesis really cannot be analyzed for consistency and convergence in the same manner as traditional numerical techniques because they are not based on the same set of fluid conservation equations. Consequently, it is believed that an absolute proof of the consistency or convergence of the numerical methods that have been developed here cannot be determined by simply expanding the difference approximations to the equations in a Taylor series and showing that they converge to the true differential equations as the mesh spacing approaches zero, since the equations used by COBRA-IIIC are derived from a control volume approach and there has never been any proof given that these equations are numerically or physically equivalent to the true differential equations whose consistency and stability have been proven in other branches of fluid mechanics. Thus, it is believed that it is first necessary to show that the COBRA-IIIC conservation equations can be derived from first principles before any meaningful proof of the convergence of the methods that have been developed for solving them can be given. Until this is done, the only way to determine the convergence of these methods is to check them with respect to computational consistency, and to compare their predictions to the limited amount of experimental data that is available. Finally, it must

be recognized that considerable uncertainty currently exists in the values of many of the empirical correlations and coupling coefficients used in the equations, and that it is conceivable that almost any method in use today could be made to agree reasonably well with experimental data by simply altering the values of these coefficients within a reasonable range to suit the characteristics of the problem on hand. Numerical proofs of the convergence of these methods will be presented for certain types of simple problems in Chapter 7.

CHAPTER VII

7.1 Numerical Tests and Results

Although a considerable amount of time has been devoted in this thesis to explaining the physical and numerical formalism behind the methods that have been discussed, the only way to demonstrate their feasibility on a practical scale is to run a series of numerical tests in which they are compared to one another and to the results of physical experiments. Consequently, an attempt will be made to present a general picture of the similarities and differences between the predictions of the methods, and to suggest specific types of experiments or sample problems where a meaningful comparison of the capabilities of these methods can be made.

7.2 Sample Problems to be Considered

Because it has been found from experience that it is extremely difficult, if not impossible, to draw any meaningful conclusions about the results of the methods when they are applied to very large and complex problems, it is believed that a systematic comparison of the similarities and differences between the methods can be best illustrated by considering simple problems with a very small number of computational cells. Consequently, almost all the results that will be presented here will be for cases

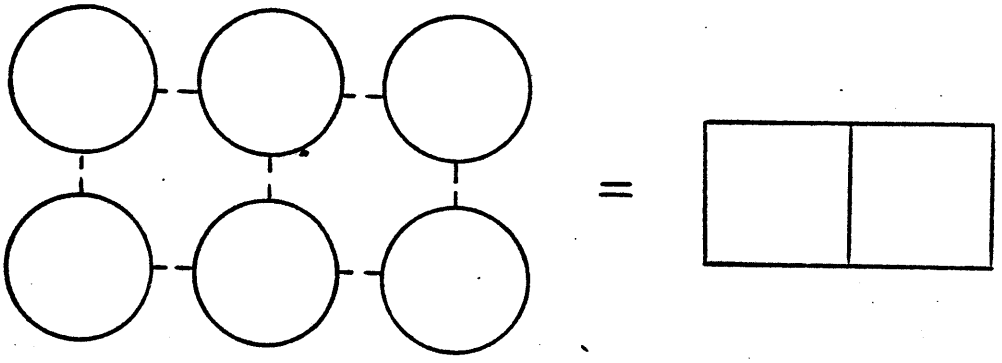
similar to those shown in Fig. 7.1 in which two adjacent computational cells representing either subchannels or fuel assemblies are connected to one another by a single common geometrical boundary so that the interchange of mass, energy, and momentum is allowed to occur. More complex cases can be considered by running the codes after the results of these simple cases are more clearly understood.

7.3 Comparisons of the Results of the Methods for Cases with Only One Axial Iteration

7.3.1 Time Step Sensitivity Studies

Since the COBRA-IIIP/MIT code is based on the single pass locally iterative axial iteration scheme discussed in section 4.1.2 and the COBRA-IIIC code is based on a true multipass axial iteration scheme, a consistent comparison of the predictions of the codes and their numerical methods cannot be given for cases which require more than one axial iteration. Consequently, the discussion in this section will be restricted to the sample problem shown in Fig. 7.2 in which the flow convergence criterion is made large enough so that only one axial iteration is performed. In this problem, a grid with a loss coefficient of 2.5 is located midway between the inlet and the outlet of two identical fuel assemblies with different radial power peaking factors, and the power level in both assemblies is doubled uniformly in one

Subchannel-Sized Cells



Assembly-Sized Cells

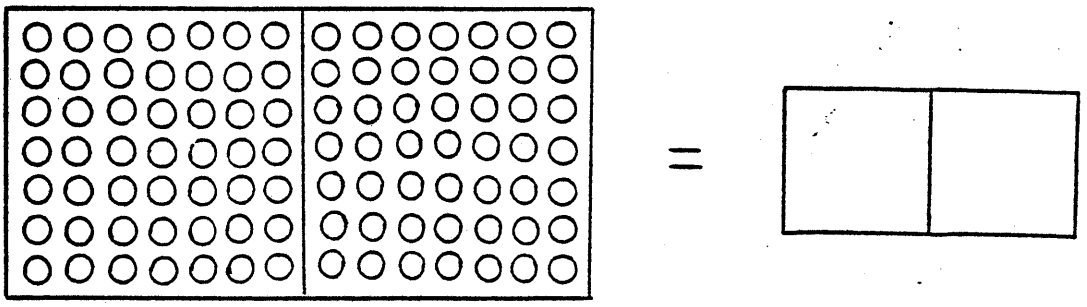


Fig. 7.1 Types of sample problems to be considered

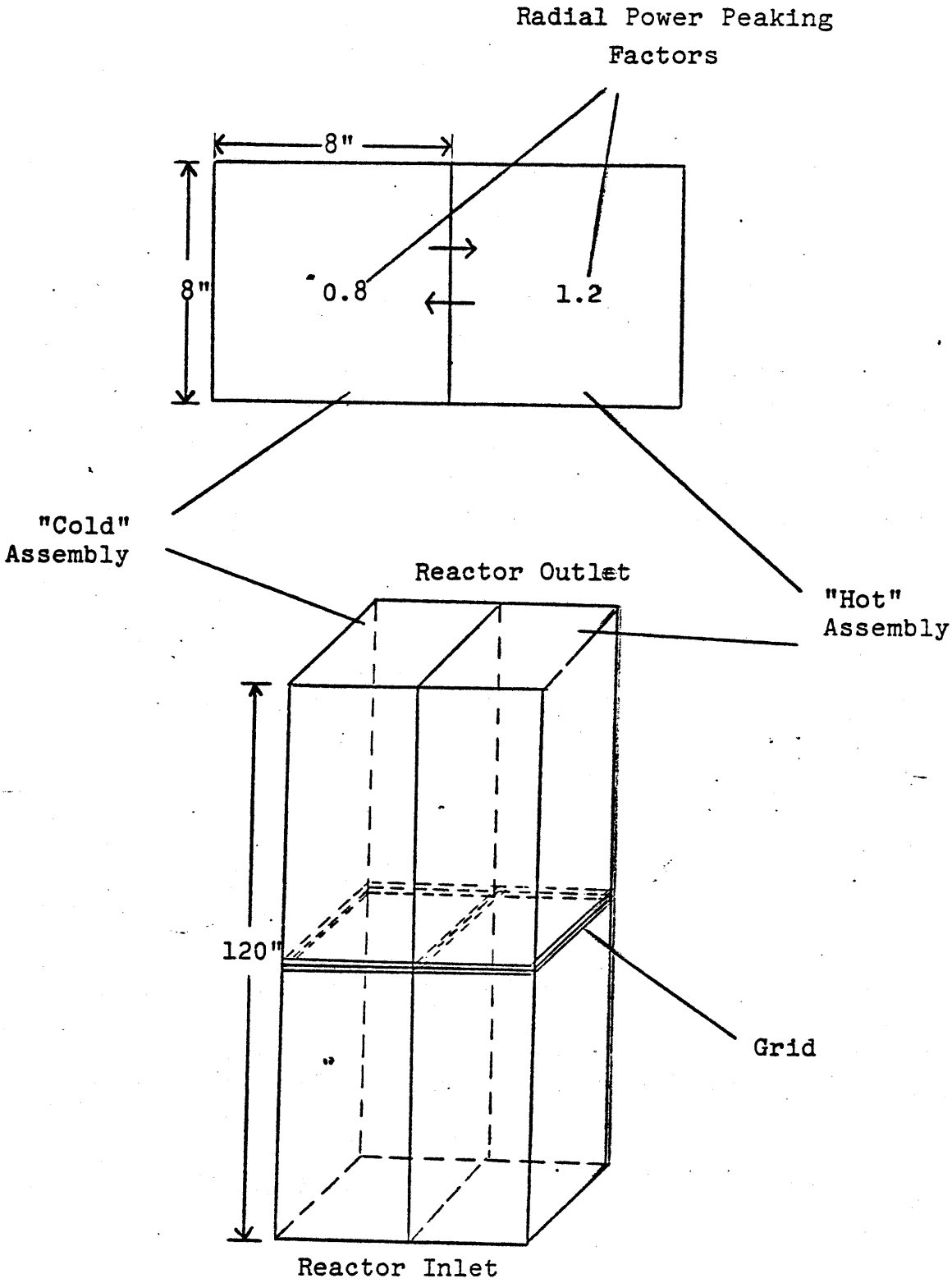


Fig. 7.2 Sample problem for time step sensitivity studies

second. With an axial mesh spacing of two inches, and with one computational cell used to represent an entire assembly in the transverse plane, a time step sensitivity study was performed in which the time step size used by the theta method, the MAT method, and the COBRA-IIIC method was varied between one second and one-tenth of a second to see the effect changing the time step size would have on the crossflow distribution. The results shown in Figs. 7.3, 7.4, and 7.5 indicate that none of these methods is very sensitive to the value of the time step size, and that the MAT method and the COBRA-IIIC method both appear to converge to approximately the same asymoptotic crossflow distribution for this problem as the time step size is decreased. This, of course, should come as no surprise because both methods solve exactly the same set of conservation equations with the same set of boundary conditions. However, the theta method, when it is used as an initial value approach, appears to converge to a crossflow distribution which is somewhat lower at the outlet of the hot channel than the distributions predicted by the other methods. This is believed to be due to the $\{a_{j-1}'\} = \{a_j'\}$ approximation discussed in section 3.1.1 and the use of a somewhat different set of boundary conditions on the inlet pressure distribution (i.e., inlet pressure = constant). Table 7.1 shows the coolant temperatures, enthalpies, axial mass flow rates, and densities predicted by all three methods at the exit of the hot channel during the course of

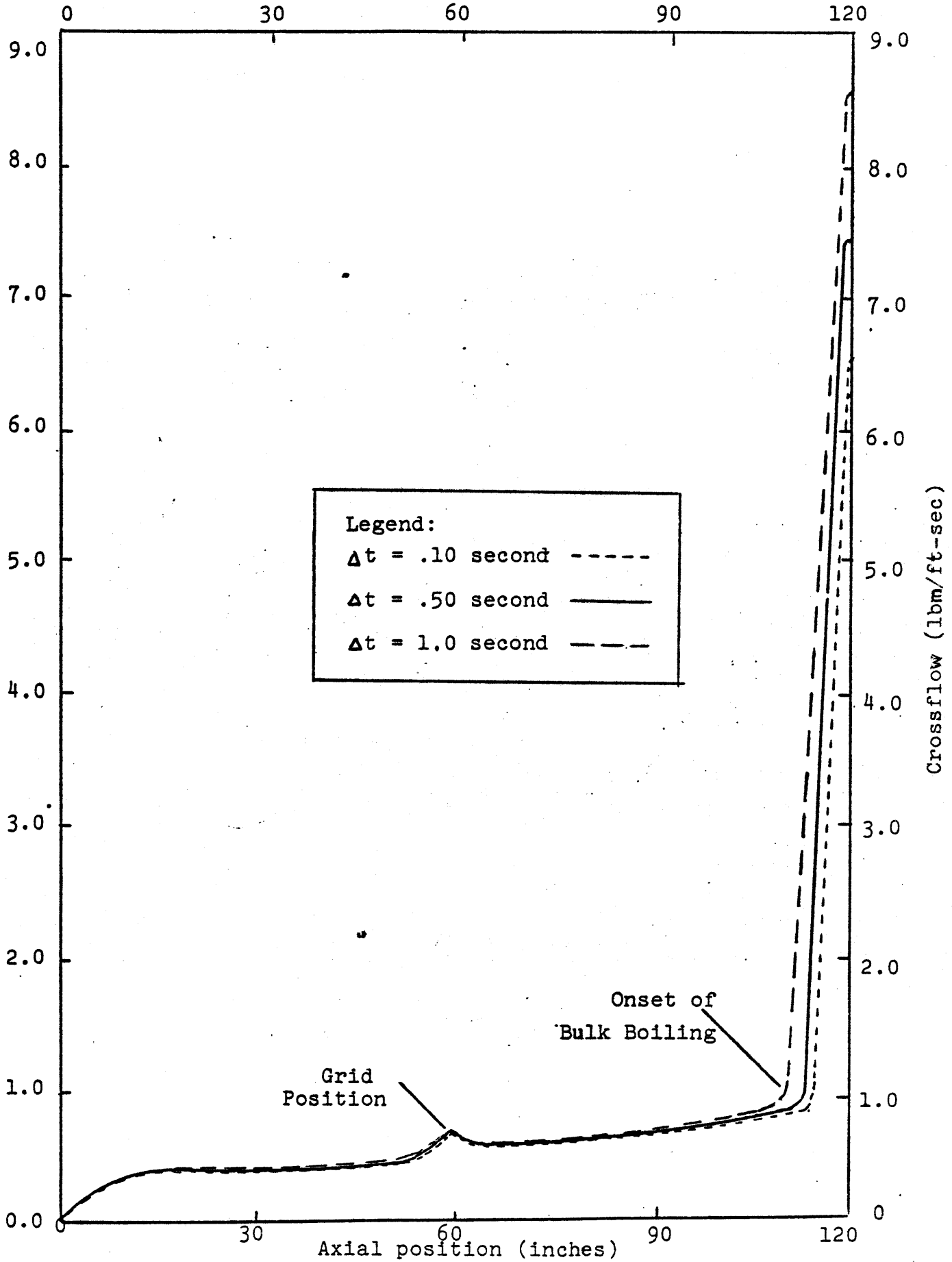


Fig. 7.3 Results of COBRA-IIIC time step sensitivity study

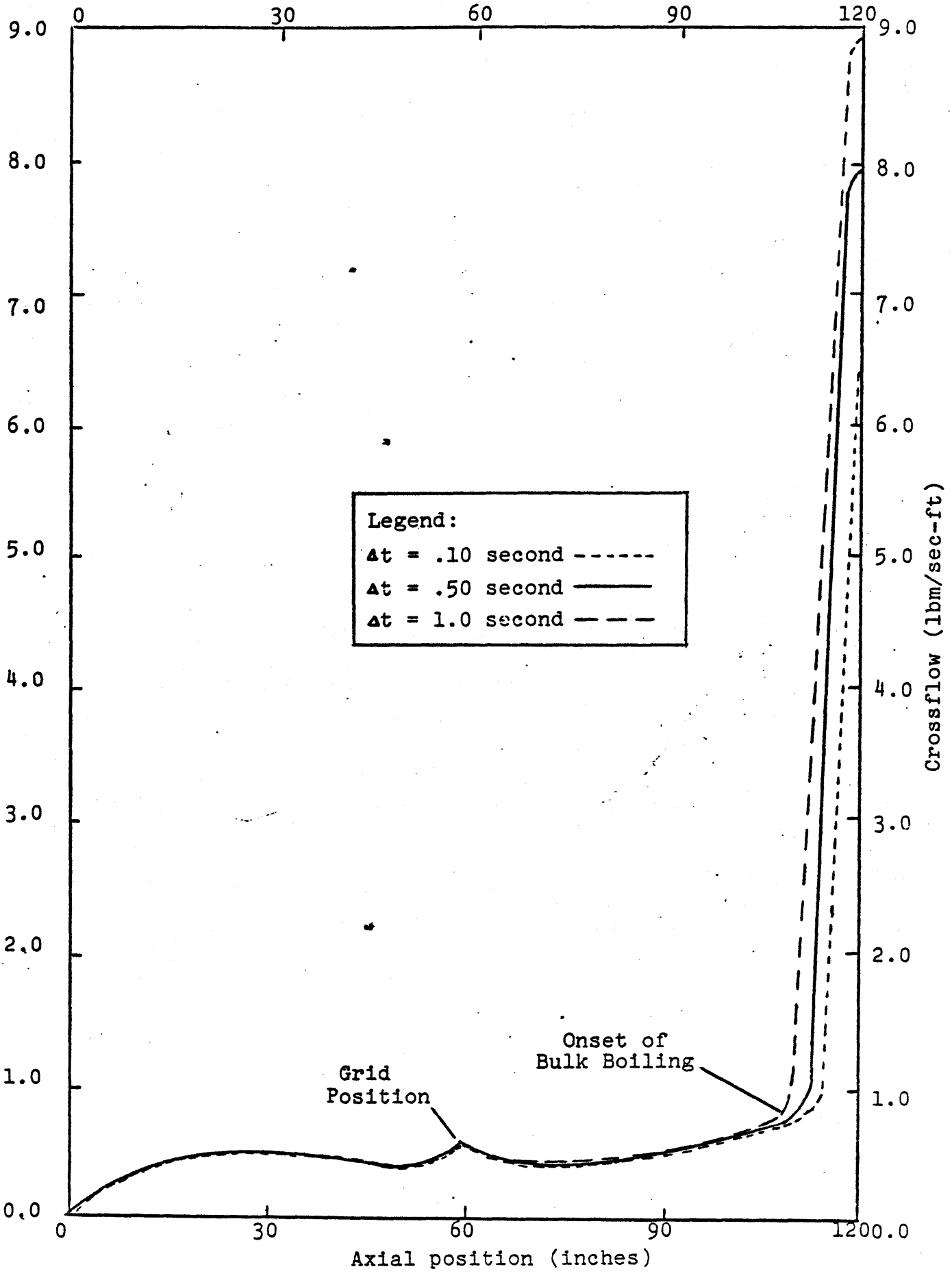


Fig. 7.4 Results of the MAT method time step sensitivity study

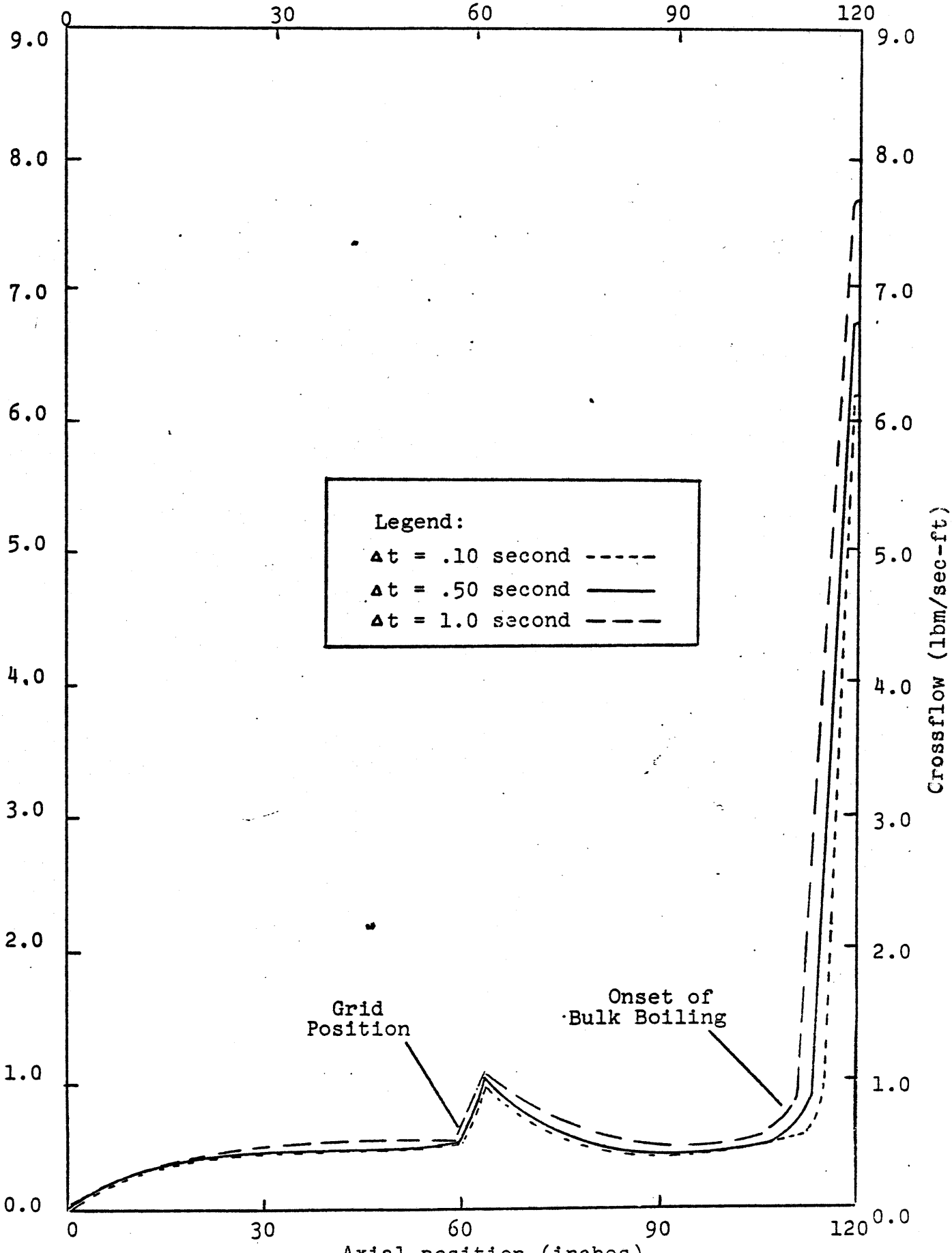


Fig. 7.5 Results of theta method time step sensitivity study

Table 7.1. Values of the Variables Predicted by Several Methods at the Outlet of the Hot Channel as a Function of Time

Time (sec)	Temperatures (F°)			Densities (lbm/ft^3)		
	COBRA-IIIC	MAT	THETA	COBRA-IIIC	MAT	THETA
0	607.30	607.33	607.32	41.65	41.65	41.65
.1	608.24	608.26	608.26	41.57	41.57	41.57
.2	610.08	610.09	610.10	41.41	41.40	41.40
.3	612.80	612.79	612.81	41.16	41.16	41.15
.4	616.29	616.27	616.30	40.84	40.84	40.83
.5	620.42	620.37	620.44	40.44	40.45	40.43
.6	625.02	624.97	624.06	39.99	39.99	39.98
.7	629.96	629.89	630.03	39.48	39.49	39.46
.8	635.08	635.01	635.14	38.93	38.94	38.90
.9	640.29	640.21	640.38	38.35	38.36	38.32
1.0	642.81	642.80	642.89	35.93	35.99	35.88

(continued on next page)

Table 7.1. (Continued)

Time (sec)	Enthalpies (BTU/lbm)			Mass Flux (Mlbm/hrft^2)		
	COBRA-IIIC	MAT	THETA	COBRA-IIIC	MAT	THETA
0	627.70	627.76	627.74	2.464	2.462	2.462
.1	629.08	629.13	629.11	2.479	2.480	2.478
.2	631.84	631.86	631.85	2.490	2.493	2.489
.3	635.91	635.90	635.91	2.499	2.501	2.493
.4	641.21	641.17	641.26	2.505	2.506	2.496
.5	647.56	647.49	647.65	2.509	2.510	2.502
.6	654.78	654.69	654.86	2.511	2.513	2.503
.7	662.67	662.57	662.79	2.511	2.513	2.504
.8	671.06	670.94	671.27	2.510	2.512	2.502
.9	679.78	679.65	679.88	2.508	2.516	2.499
1.0	688.73	688.60	688.84	2.501	2.509	2.492

the transient. Although it can be seen that all three methods predict slightly different magnitudes and shapes for the cross flow distribution as a function of time, these differences appear to have a very small (and almost negligible) effect upon the properties of the fluid at the outlet of the hot channel. In fact, the maximum difference between the predictions of the methods at any time during the transient is less than two tenths of one per cent. Since this is at least one or two orders of magnitude less than the uncertainties that are inherent in some of the empirical correlations used by the codes to compute the heat transfer coefficients and the axial pressure drop, this example serves to illustrate the point that it makes very little difference from a practical point of view which method is used to analyze the effects of a moderately severe transient as long as the crossflow is less than a few per cent of the axial mass flow rate. For transients where much more severe boiling takes place over a larger fraction of the axial length of the core, or where the effects of flow blockages must be taken into account, the $\{a_{j-1}'\} = \{a_j\}$ approximation used by the theta method ceases to be valid, and considerable differences may develop between its predictions and those of the other methods. For example, if the radial power peaking factors in the assemblies are changed to 1.5 and 0.5 to create a 3 to 1 power gradient across the bundles, the crossflow distribution predicted

by the theta method at the end of the transient is considerably different than that of the other two methods, as illustrated in Fig. 7.6. This slightly different sample problem again serves to illustrate the point that the MAT method and the method used in COBRA-IIIC give results that are in reasonably good agreement with one another as long as only one axial iteration is performed. However, it must be recognized that a hypothetical experiment for this problem which had an error of only 5% or 10% in the measured values of the axial mass flow rates could not possibly distinguish between the predictions of the MAT method and COBRA-IIIC and probably could not even determine if the $\{a_{j-1}'\} = \{a_j'\}$ approximation used by the theta method was a valid assumption for this case.

7.3.2 Space Step Sensitivity Studies

In order to demonstrate the sensitivity of the methods that have been discussed to changes in the size of the axial mesh spacing, a sensitivity study was performed in which the mesh spacing was varied between one inch and one foot for the problem that was described previously. As shown in Figs. 7.7, 7.8, and 7.9 the COBRA-IIIC method, the theta method, and the MAT method were all found to converge rather rapidly to approximately the same asymoptotic axial mass flow distribution as the mesh spacing was reduced. These results apparently indicate that none of these methods is very sensitive to the size of the axial mesh spacing, and that for many practical problems, an axial mesh spacing

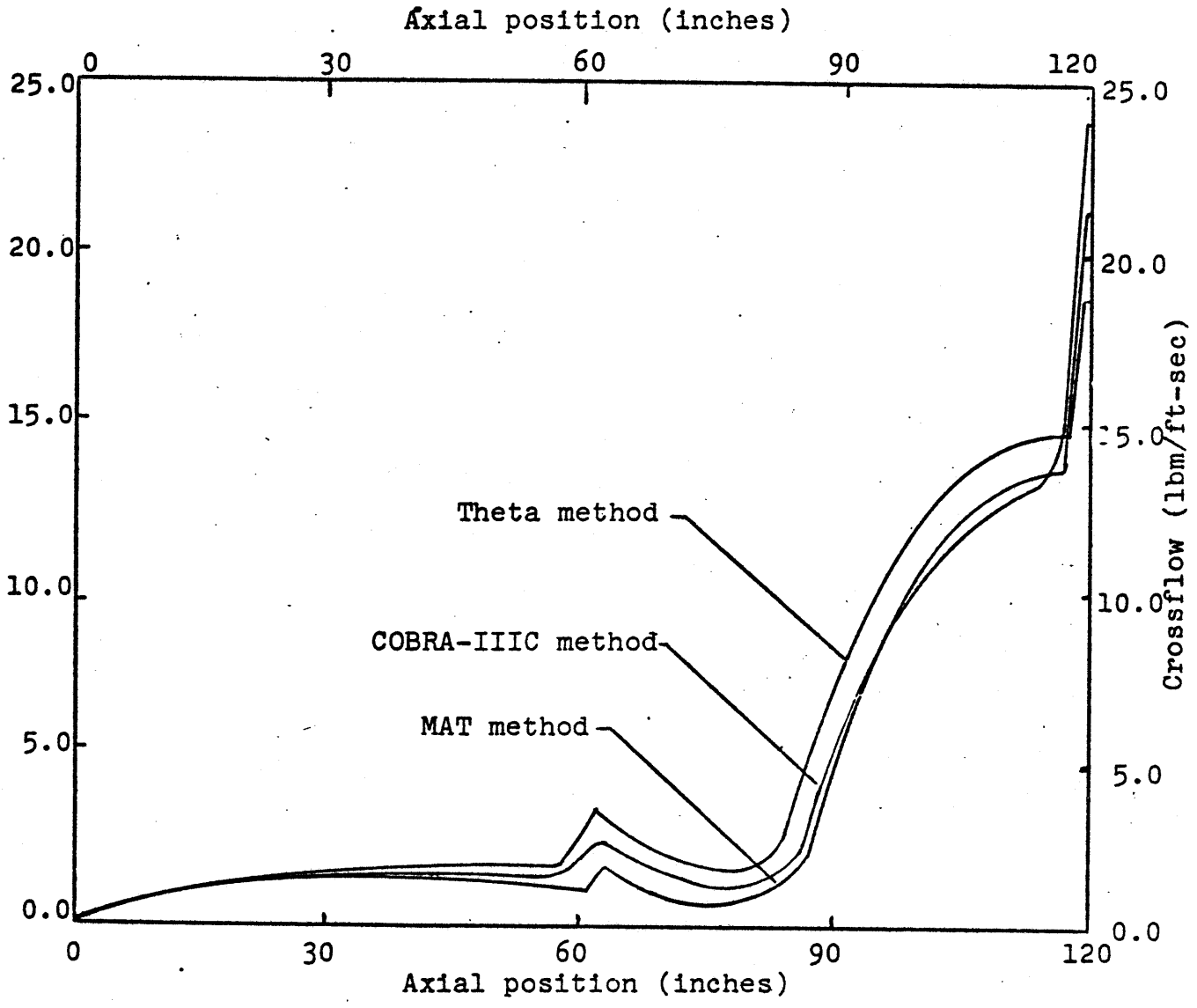


Fig. 7.6 Crossflow distributions predicted for a 3 to 1 radial power gradient

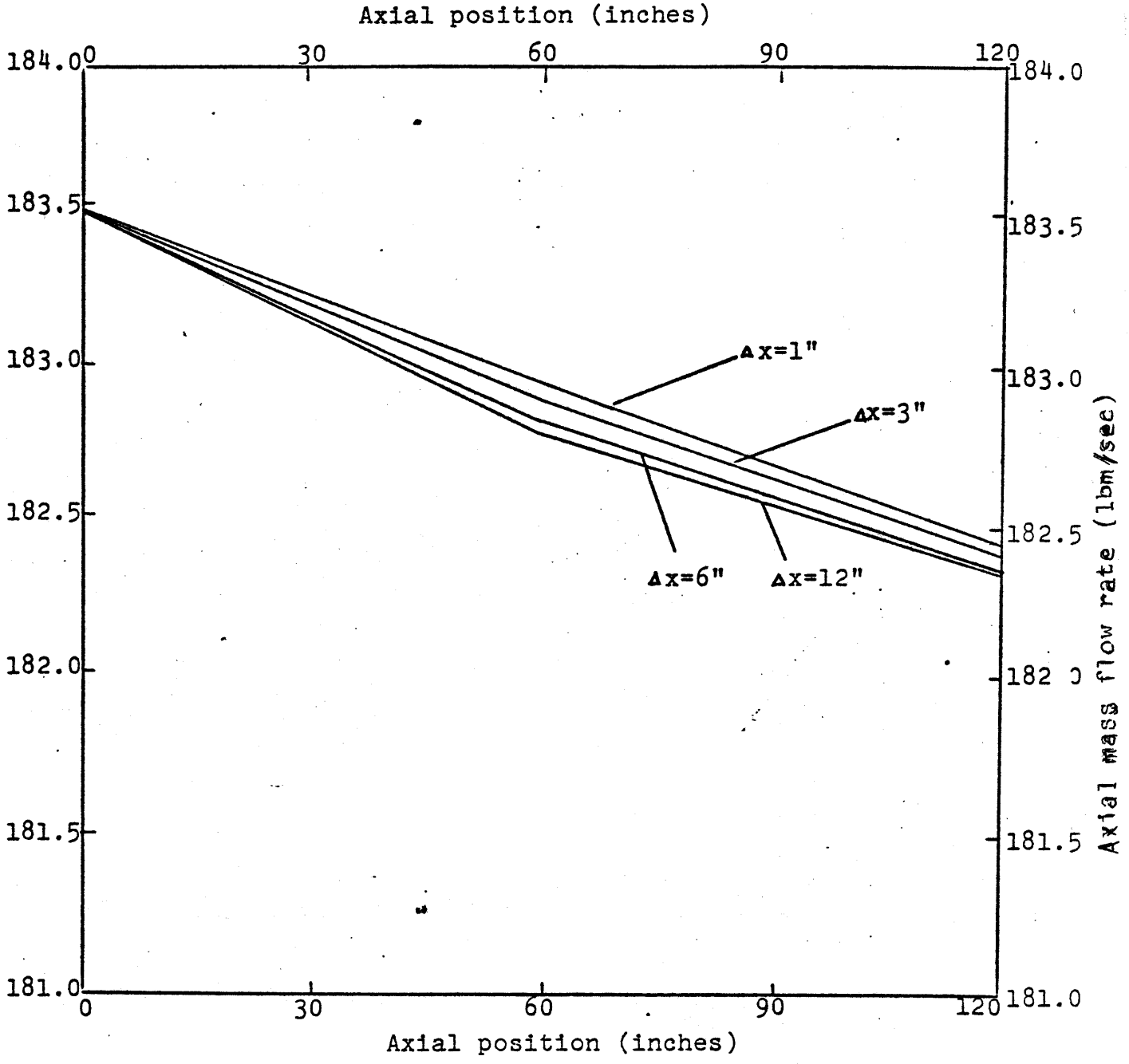


Fig. 7.7 Results of COBRA-IIIC Space Step Sensitivity Study

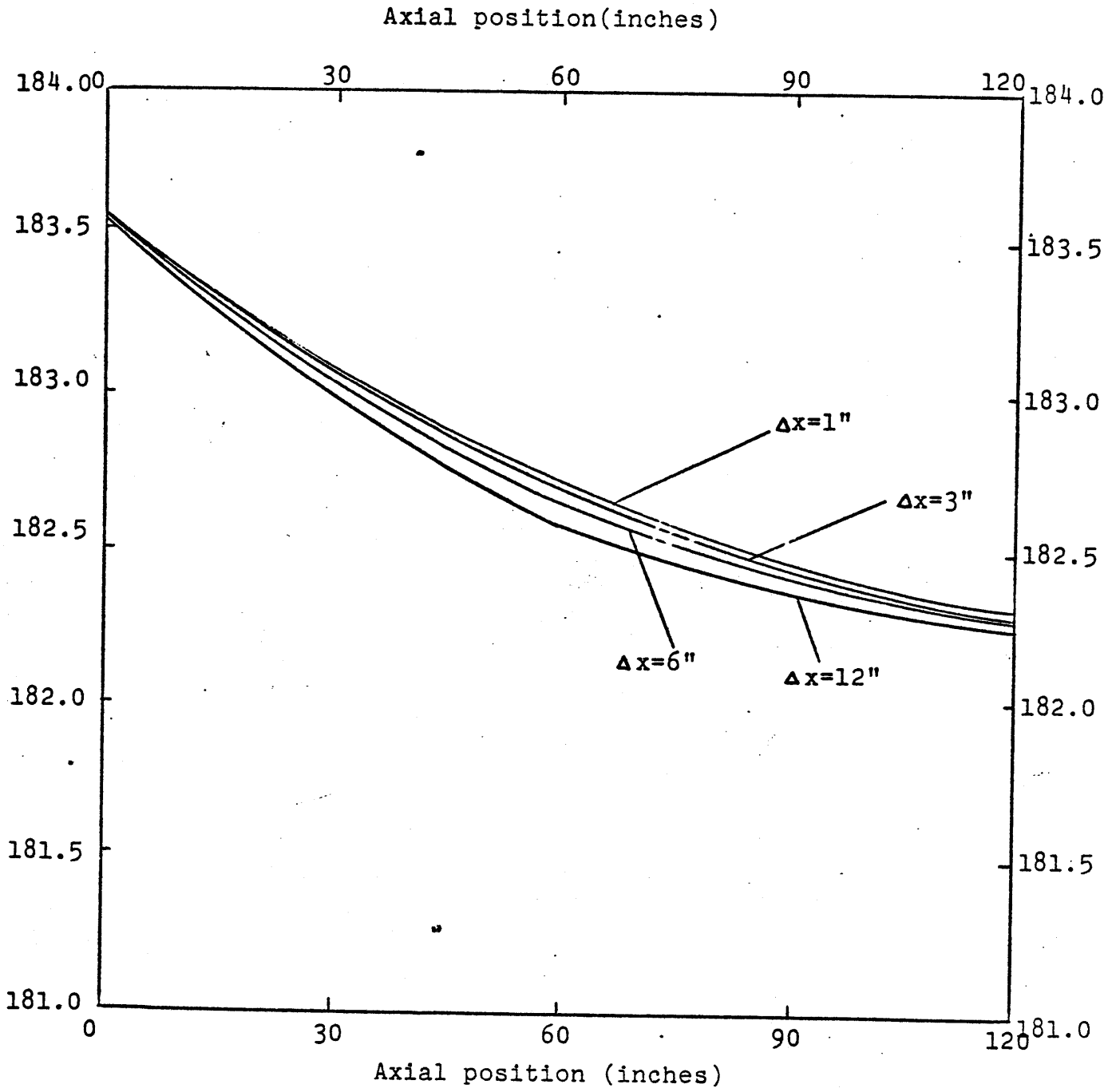


Fig. 7.8 Results of the MAT Method Space Step Sensitivy Study

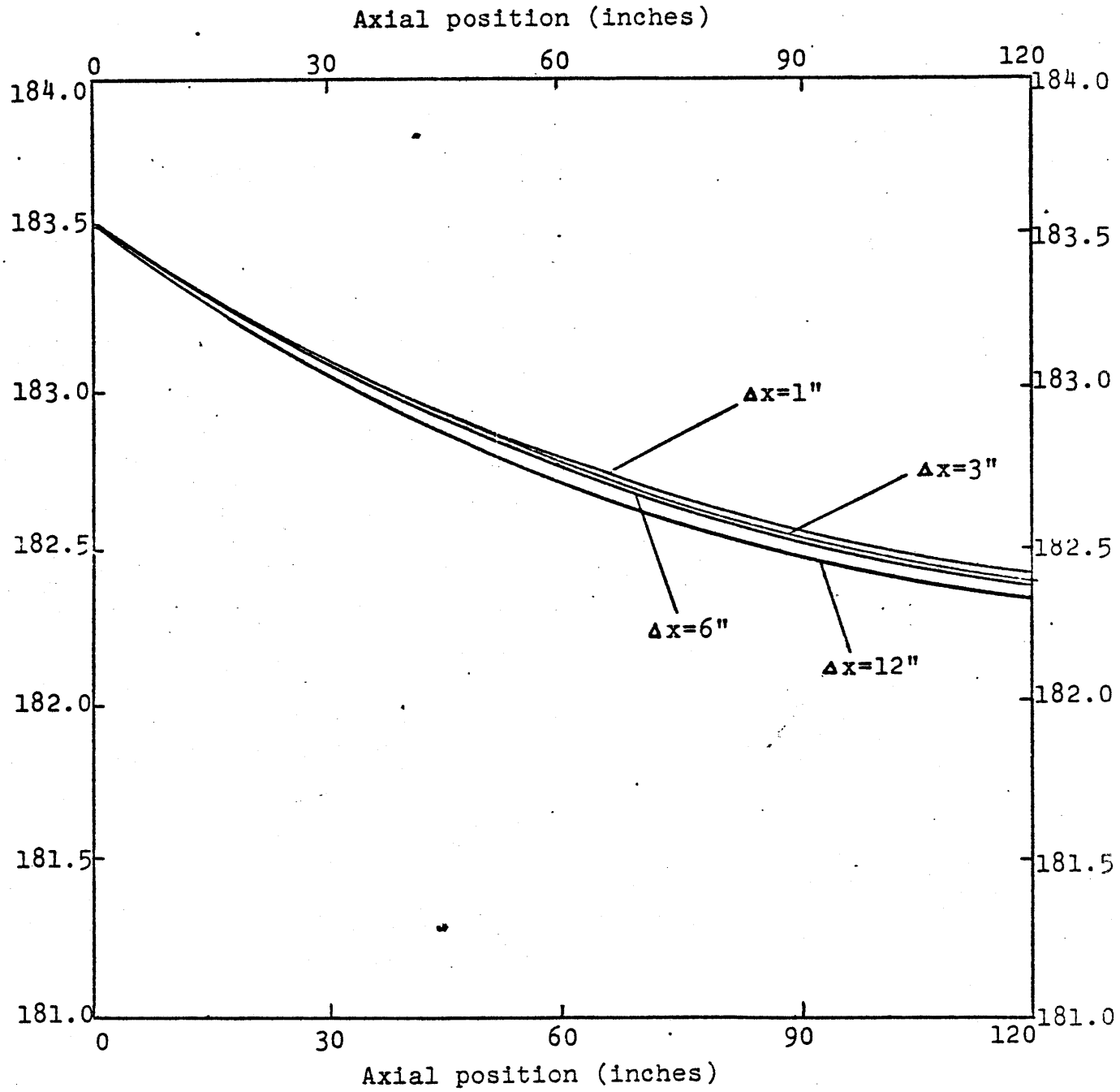


Fig. 7.9 Results of the Theta Method Space Step Sensitivity Study

of six inches or one foot is sufficient to give an acceptable estimate of the axial mass flow rate.

Reducing the axial mesh spacing apparently reduces the errors in the estimates of $\{m_j\}$ that are used to compute the crossflow distribution for the first iteration, and presumably causes the methods to converge to an "asymoptotic" distribution. However, it is not clearly understood why the MAT method and COBRA-IIIC should not converge to <u>exactly</u> the same answers as the axial mesh spacing is reduced, since they attempt to solve exactly the same system of difference equations when θ is set equal to 0. It is conceivable that these slight differences for the first axial iteration may be due to roundoff errors, since all the computations in COBRA-IIIC are carried out in single precision, whereas most of the computations in COBRA-IIIP/MIT are performed in double precision. Again it must be recognized that the differences illustrated here are no larger than a few tenths of a per cent, and are so small that they cannot possibly be resolved by experimental measurement. Thus the "accuracy" of the codes for problems of this kind can only be judged by making an objective evaluation of the physical consistency of their results, unless analytic solutions to the conservation equations for meaningful problems are available.

7.4 Effect of the Axial Iteration Scheme on the Solution

In previous chapters of this thesis it was shown that the COBRA-IIIC code is based on a multipass axial iteration scheme which allows information from downstream locations to be propagated upstream at the rate of one axial node per iteration. However, no analytical proof of the stability, the consistency, or the convergence of this scheme has ever been given, and it has never been determined whether additional iterations always improve the accuracy of the solutions. The purpose of this section is to demonstrate that the axial iteration scheme used by COBRA-IIIC does not always appear to improve the accuracy of the solutions to the conservation equations, and in some cases, actually appears to make the results considerably worse because it introduces strange oscillations into the shape of the cross-flow distribution which cannot be explained physically. The effect that these oscillations can have on the predictions of the code will generally be shown to be small, although they can have an extremely important effect on the total computation time.

Fig. 7.10 shows a very simple problem that will be used to test the consistency of the axial iteration scheme contained within the code. This problem consists of two identical fuel pin bundles from a PWR whose size and operational characteristics are described in Table 7.2. One computational cell is used to represent each assembly and both assemblies

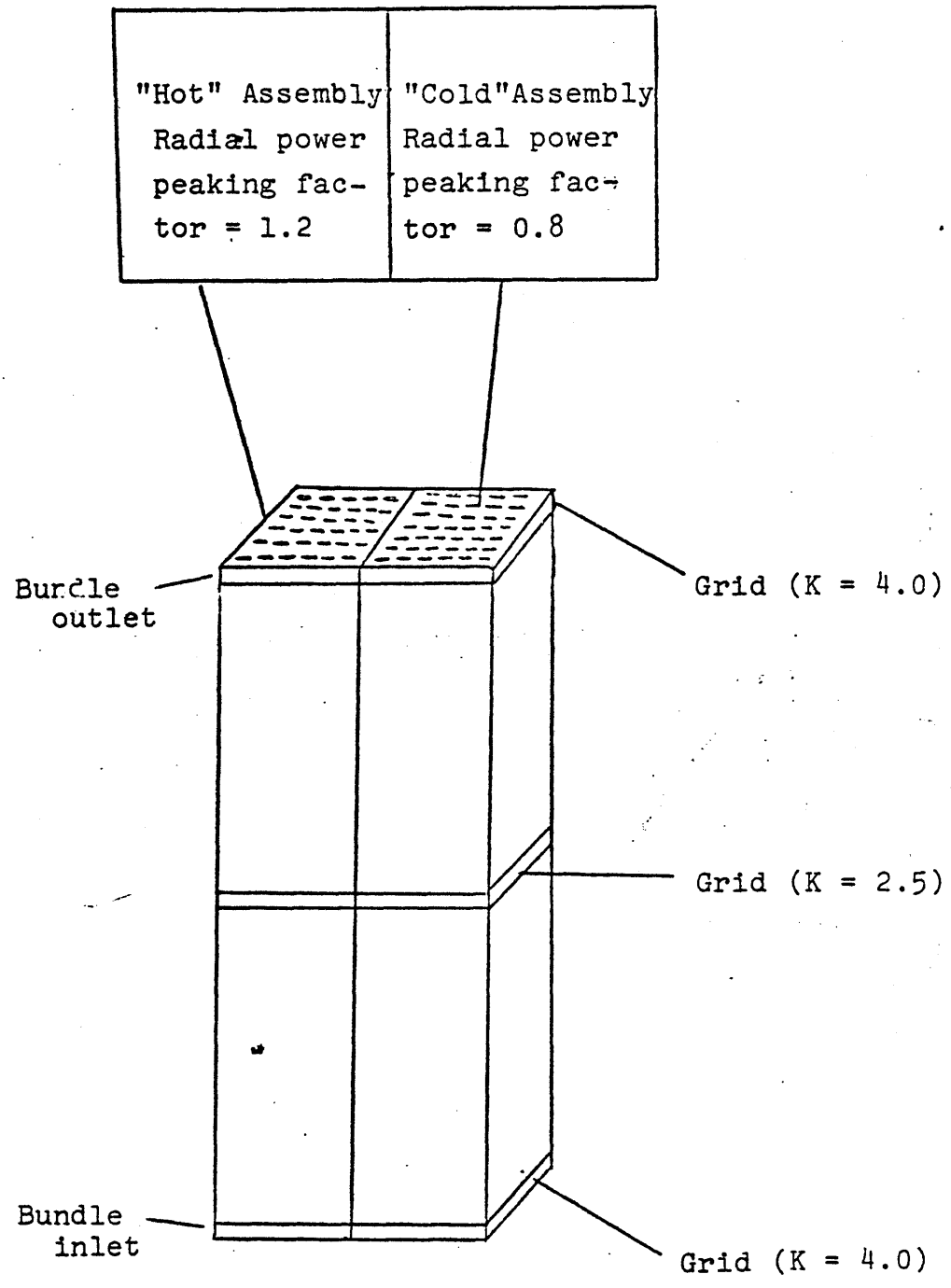


Fig. 7.10 Sample Problem for Testing the Effects of the Axial Iteration Scheme

Table 7.2. PWR Fuel Bundle Characteristics for a Sample Problem Used to Test the Axial Iteration Scheme

PWR Fuel Assemblies	
Axial power distribution - uniform Radial power peaking factors - 0.8 and 1.2 Number of cells per assembly - 1	
Nominal operating conditions:	
System outlet pressure:	2100 psia
Inlet enthalpies: (uniform for all channels)	538 BTU/lbm
Average inlet mass flux: (uniform for all channels)	2.48×10^6 lbm/hr/ft^2
Average assembly heat flux:	$.2 \times 10^6$ BTU/hr/ft^2
Channel length:	120 inches
Axial mesh spacing:	2 inches
Flow area for each cell:	.267 ft^2
Number of rods per cell:	225

are subjected to a loss of flow transient in which the mass flow rate into each assembly is reduced uniformly by a factor of 2 in two seconds. This problem was run twice with the COBRA-IIIC code with exactly the same input data, but with two different values -.01 and .001- for the flow convergence criterion. The crossflow distributions predicted by the code for these convergence criteria, and the number of axial iterations that were actually performed as a function of time are shown in Figs. 7.11, 7.12, and 7.13. It can clearly be seen that the code does not appear to converge to an asymoptotic flow distribution as the convergence criterion is tightened and more axial iterations are performed. In fact, these figures apparently show that it tends to diverge and develop more unphysical oscillations as the number of axial iterations is increased. The version of the MAT method used in COBRA-IIIP/MIT apparently does not suffer from this problem because it is programmed into the code with a single pass, locally iterative axial iteration scheme. The crossflow distribution predicted for this problem by the MAT method is shown in Figs. 7.14, 7.15, and 7.16. It can be seen that the MAT method converges very rapidly to an asymoptotic flow distribution during each step in the transient as more iterations are performed. Thus it is believed that the version of the MAT method used by COBRA-IIIP/MIT gives a more consistent picture of what one would expect to happen physically when it is applied to this simple problem.

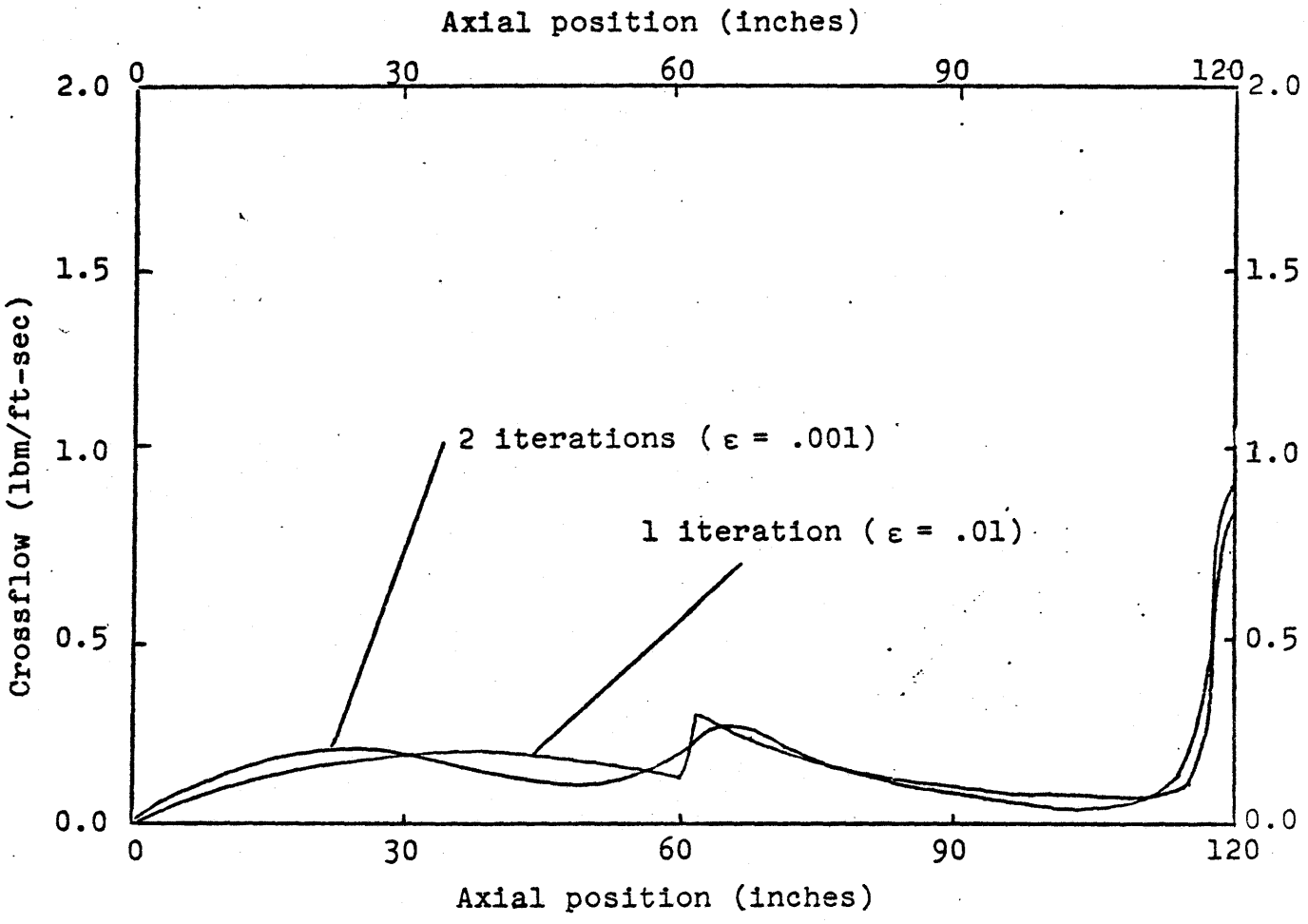


Fig 7.11 Effects of the COBRA-IIIC Axial Iteration Scheme at t = 0 Seconds

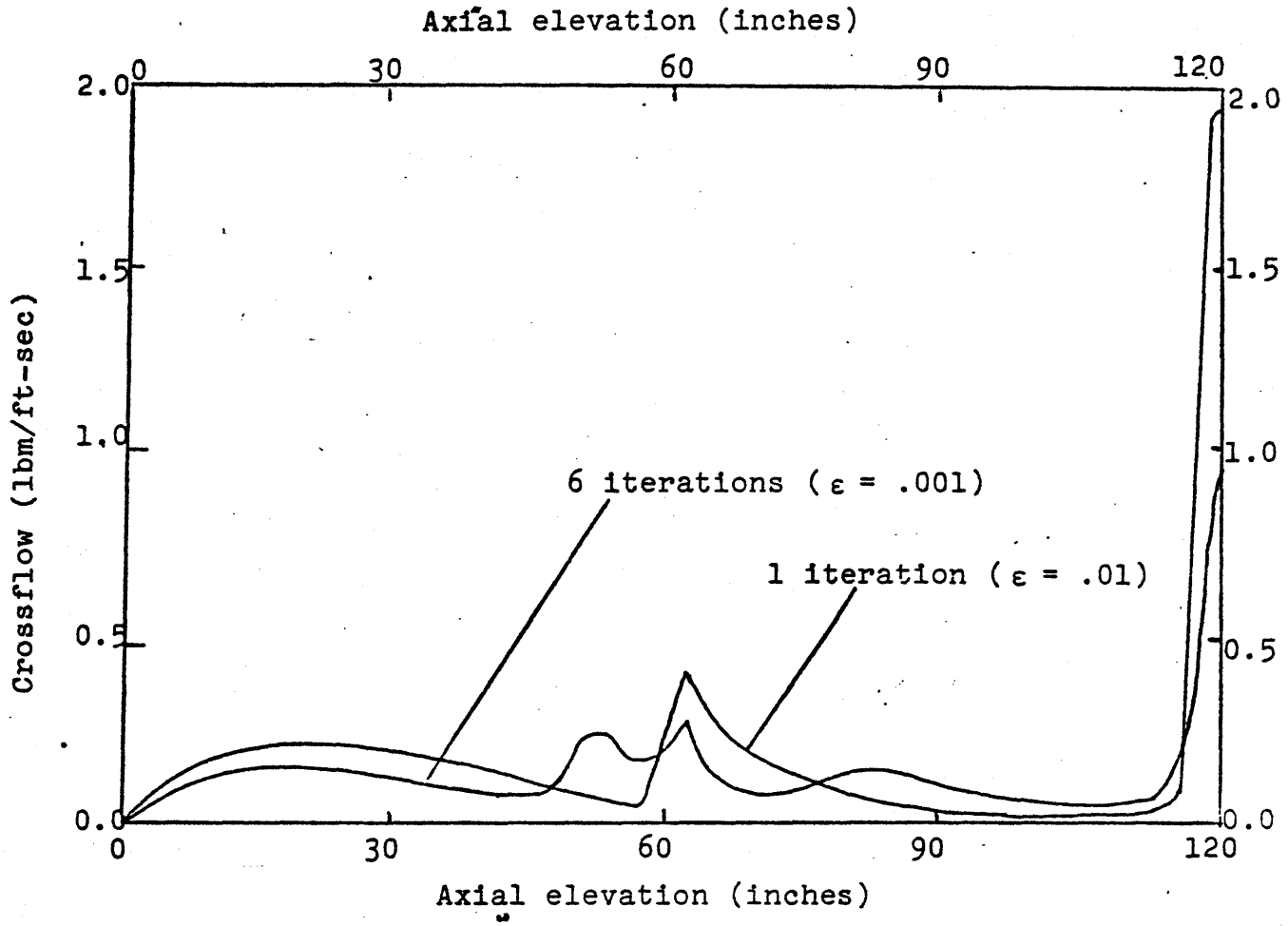


Fig. 7.12 Effects of the COBRA-IIIC Axial Iteration Scheme at t = 1 Second

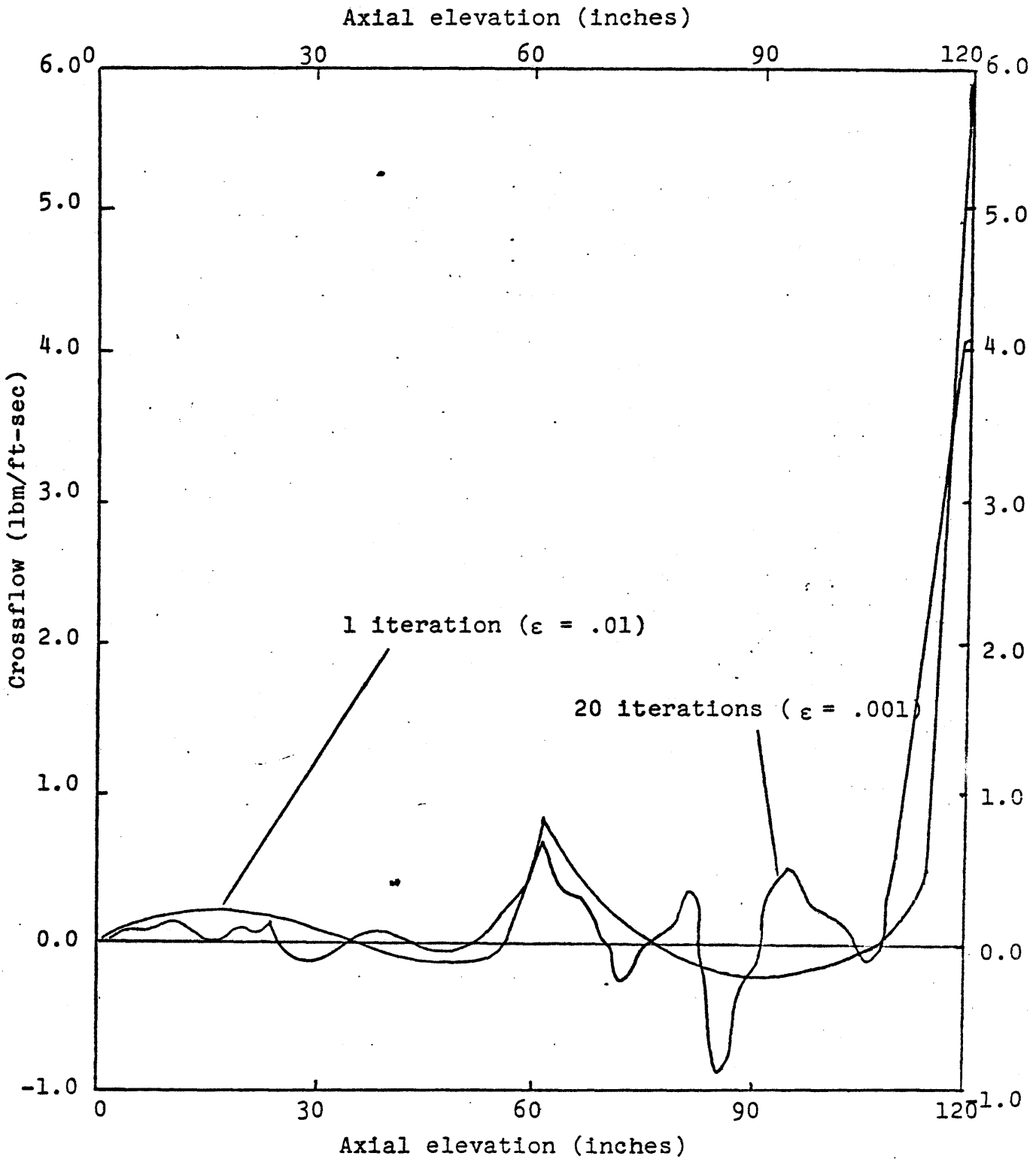


Fig. 7.13 Effects of the COBRA-IIIC Axial Iteration Scheme at t = 2 Seconds

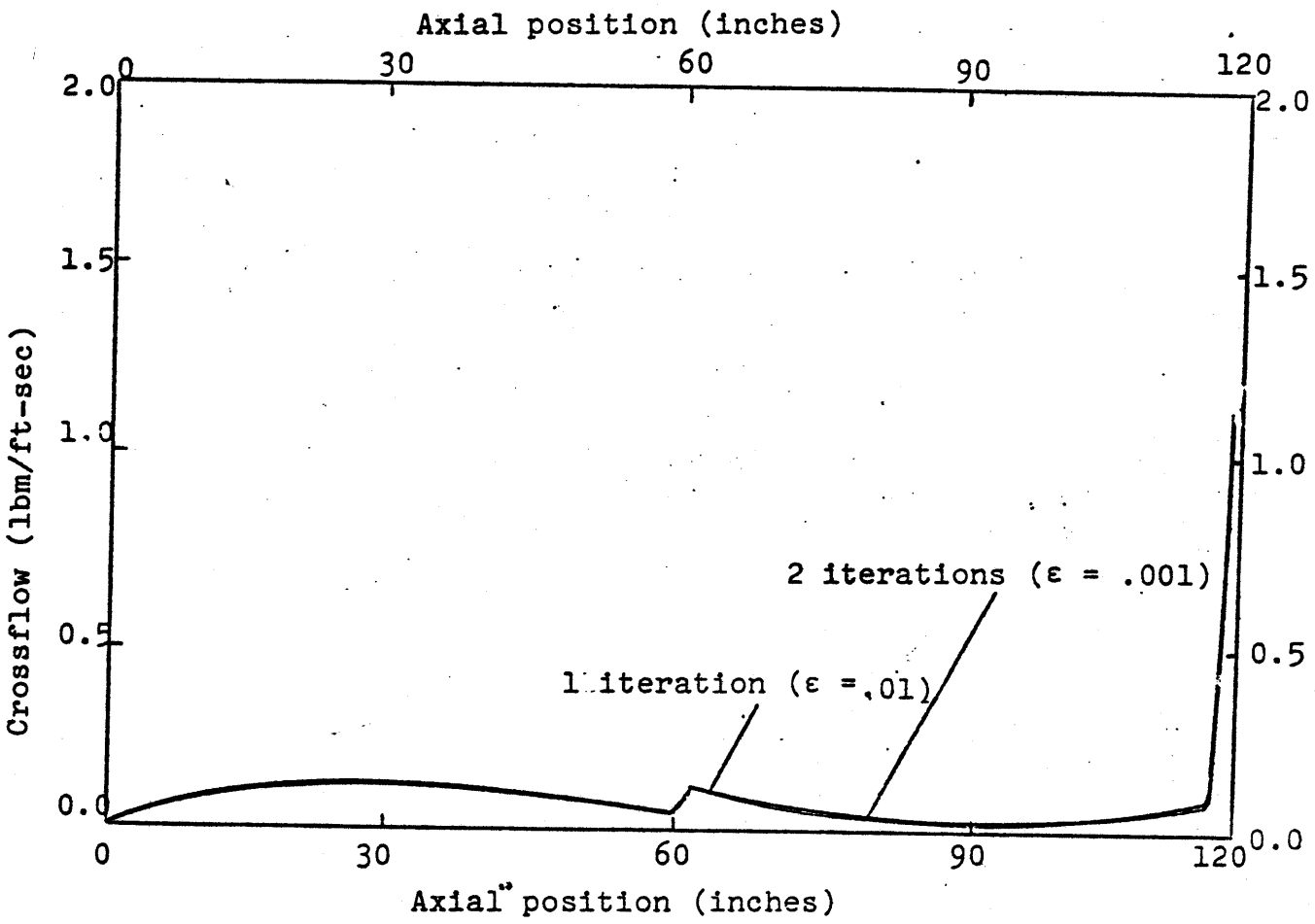


Fig. 7.14 Effects of the MAT Method Axial Iteration Scheme at t = 0 Seconds

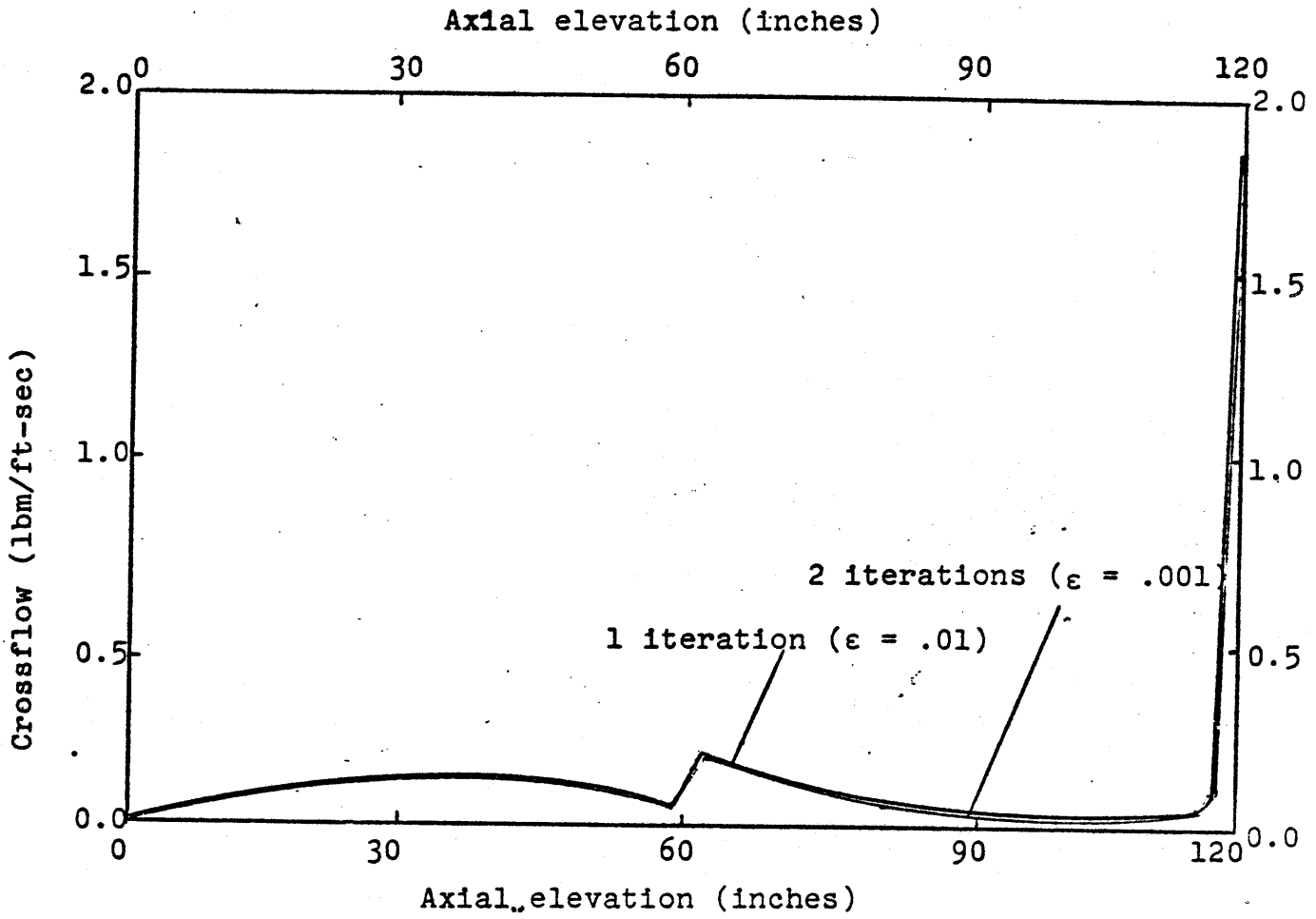


Fig. 7.15 Effects of the MAT Method Axial Iteration Scheme at t = 1 Second

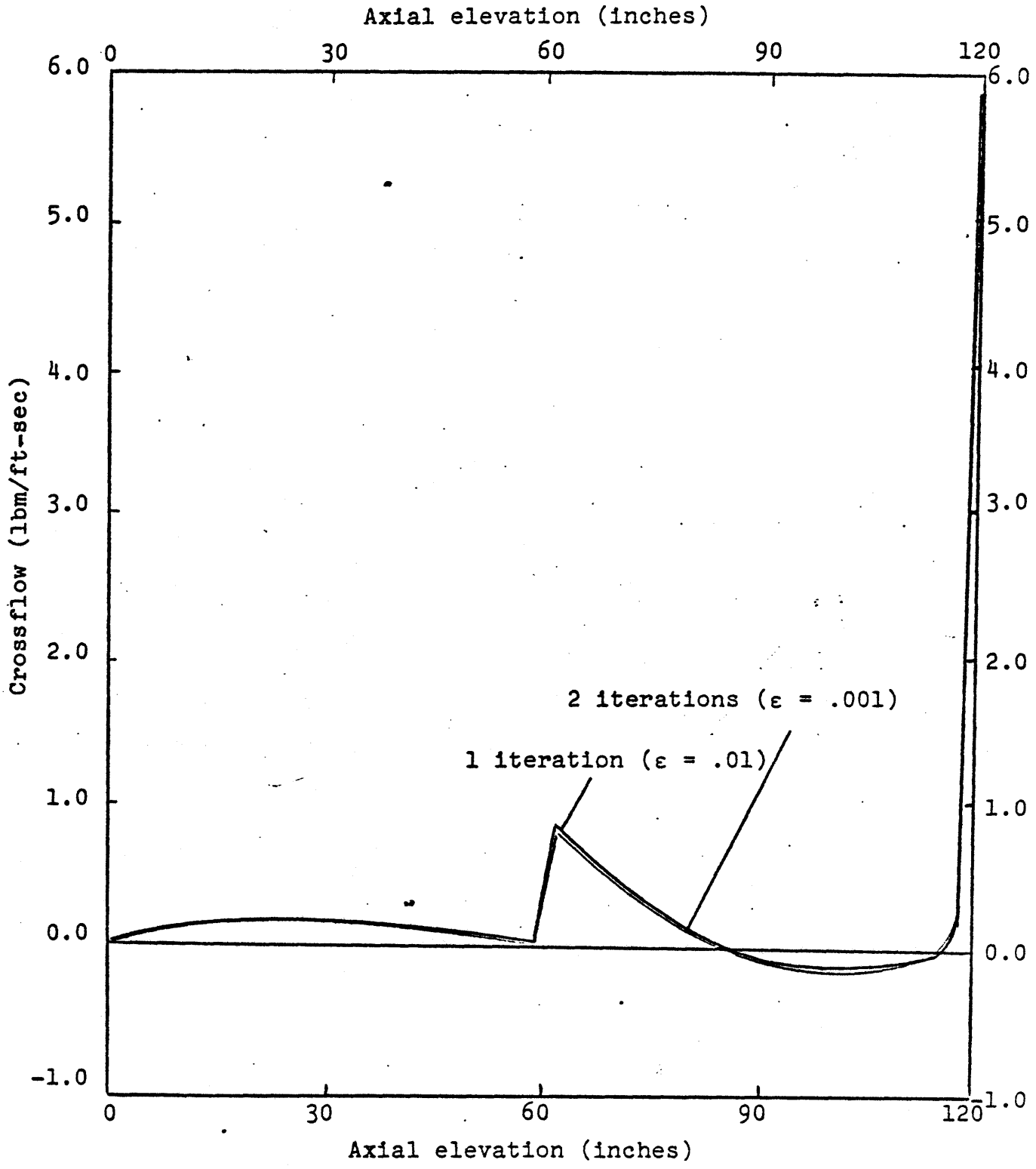


Fig. 7.16 Effects of the MAT Method Axial Iteration Scheme at t = 2 Seconds

7.5 Effect of the Axial Iteration Scheme on the Axial Mass Flow Rate and Other Important Variables

Since it has been shown that performing more axial iterations with COBRA-IIIC does not necessarily improve the accuracy of the code's solutions to the fluid conservation equations, it is extremely important to determine the effect that the oscillations developed by the iteration scheme can have on the code's estimates of variables such as the axial mass flow rate, the temperature, the density, and the enthalpy of the coolant. To illustrate the magnitude of this effect, the values of these variables at the outlet of the hot assembly of the same problem were studied as a function of the number of axial iterations that were performed. The results of this study, which are summarized in Table 7.3,appear to indicate that none of these variables is very sensitive to the details of the shape of the crossflow distribution. In fact, it has been found that the oscillations produced by the axial iteration scheme hardly show up at all in plots of the axial mass flow distribution, and that the maximum change in the values of the other variables as the result of these oscillations is less than half a percent for this case. Since these variations are much smaller than any experiment could possibly detect, it is likely that the oscillations were never reported before simply because they did not have much of an effect upon the predictions of the code. Consequently, there was no reason to look for

Table 7.3 Effect of the COBRA-IIIC Axial Iteration Scheme on the Values of Several Variables at the Outlet of the Hot Channel

CASE A	Flow Convergence Criterion = .01 (Average number of axial iterations = 1)			
Time (sec.)	Temperature (°F)	Enthalpy (BTU/lbm)	Density (lbm/ft^3)	Flow rate (lbm/sec)
0.0	607.29	627.68	41.65	182.19
1.0	620.56	647.78	40.43	137.61
2.0	642.81	688.34	36.10	92.87

CASE B	Flow Convergence Criterion = .001 (Average number of axial iterations = 8)			
Time (sec)	Temperature (°F)	Enthalpy (BTU/lbm)	Density (lbm/ft^3)	Flow rate (lbm/sec)
0.0	607.30	627.68	41.64	182.18
1.0	620.55	647.76	40.42	137.38
2.0	642.80	688.17	36.18	92.24

them and even less motivation to try to explain the reason for their existence. However, from a mathematical point of view, it is clear that these effects can have an extremely important practical implication, since they apparently indicate that there is little to be gained by performing more than one or two axial iterations with the COBRA-IIIC code if doing so simply consumes a great deal of additional computer time and causes the answers to some problems to become worse.

Although it has not been possible to explain why these effects occur physically, it is conceivable that additional axial iterations may introduce roundoff errors into the solutions which cause their accuracy and their consistency to be destroyed. Another possible explanation of this phenomenon is that the axial iteration scheme used by the code is not an appropriate model of physical reality. This is believed to be due to the fact that the code does not know inherently when to stop transferring information from downstream to upstream locations, since the distance this information is "pumped" upstream is solely a function of the flow convergence criterion one wishes to choose. Consequently, choosing a convergence criterion which is too "tight" may cause the crossflow distribution to oscillate in order to conserve mass and momentum while information about the downstream flow distribution is propagated so far upstream that it may be pumped entirely out of the core.

7.6 Comparison of the Results of the Methods for a Flow-Blockage Analysis

Although it has been found that the theta method cannot be applied to study the effects of severe flow blockages when it is formulated as an initial value approach because it tends to encounter problems with numerical stability, both the MAT method and the COBRA-IIIC method do not suffer from this drawback, and can be used to analyze a variety of problems in which severe flow blockages occur. In this section, the predictions of these two methods will be compared to one another for a simple problem that contains a flow blockage. The purpose of this section will be to show the effects the axial iteration scheme used by these methods can have upon the consistency of the results.

The test problem that will be used to make this comparison is shown in Fig.7.17. The problem consits of two identical fuel assemblies with different radial power peaking factors and a constant axial heat flux distribution. A flow obstruction in the form of a grid with a loss coefficient of 2.5 is placed across the entire hot assembly, but not across the cold assembly, at a position midway between the inlet and the outlet of the assemblies. Both assemblies are then subjected to a loss of flow transient in which the inlet mass flow rates, which are initially uniform, are reduced uniformly by a factor of 2 in two seconds. With a single computational cell used to represent each assembly, the COBRA-IIIC method and the MAT method were both used to analyze this problem with a flow convergence criterion of .05. This analysis was then repeated with a "tighter" convergence criterion of .01. Figures 7.18, 7.19,

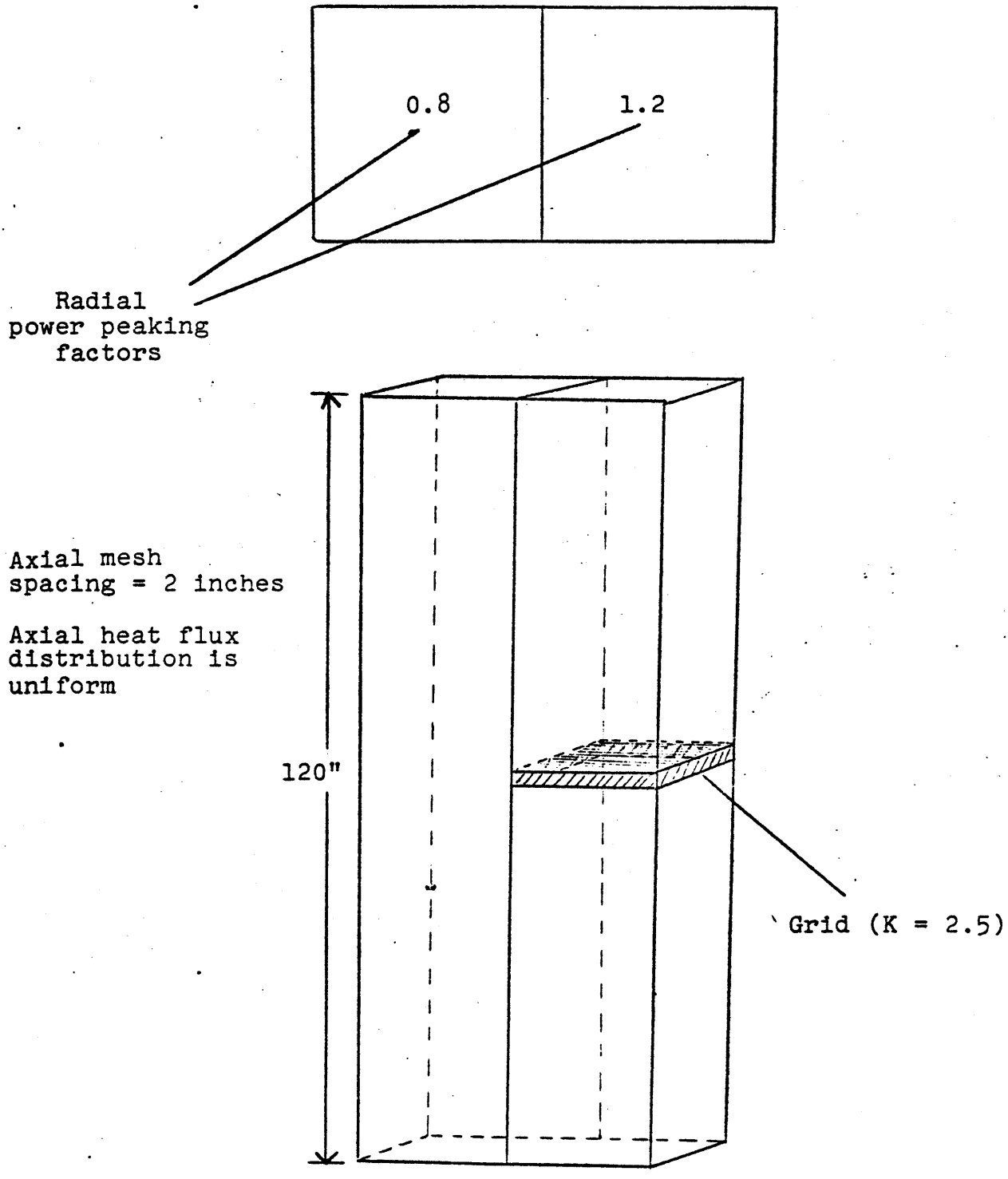


Fig. 7.17 Test Problem for Flow Blockage Calculations

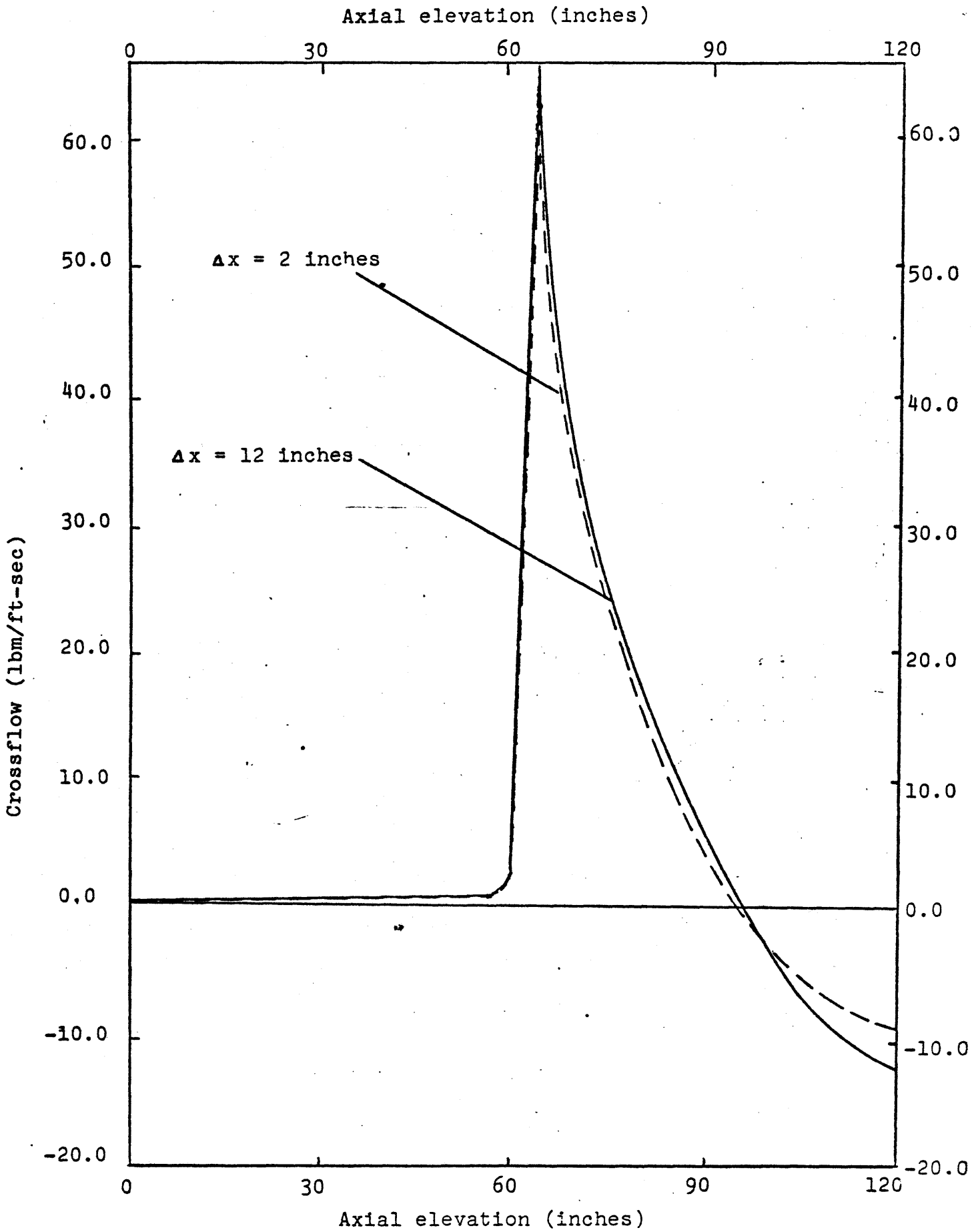


Fig. 7.18 MAT Method Blockage Predictions at t = 0 Seconds

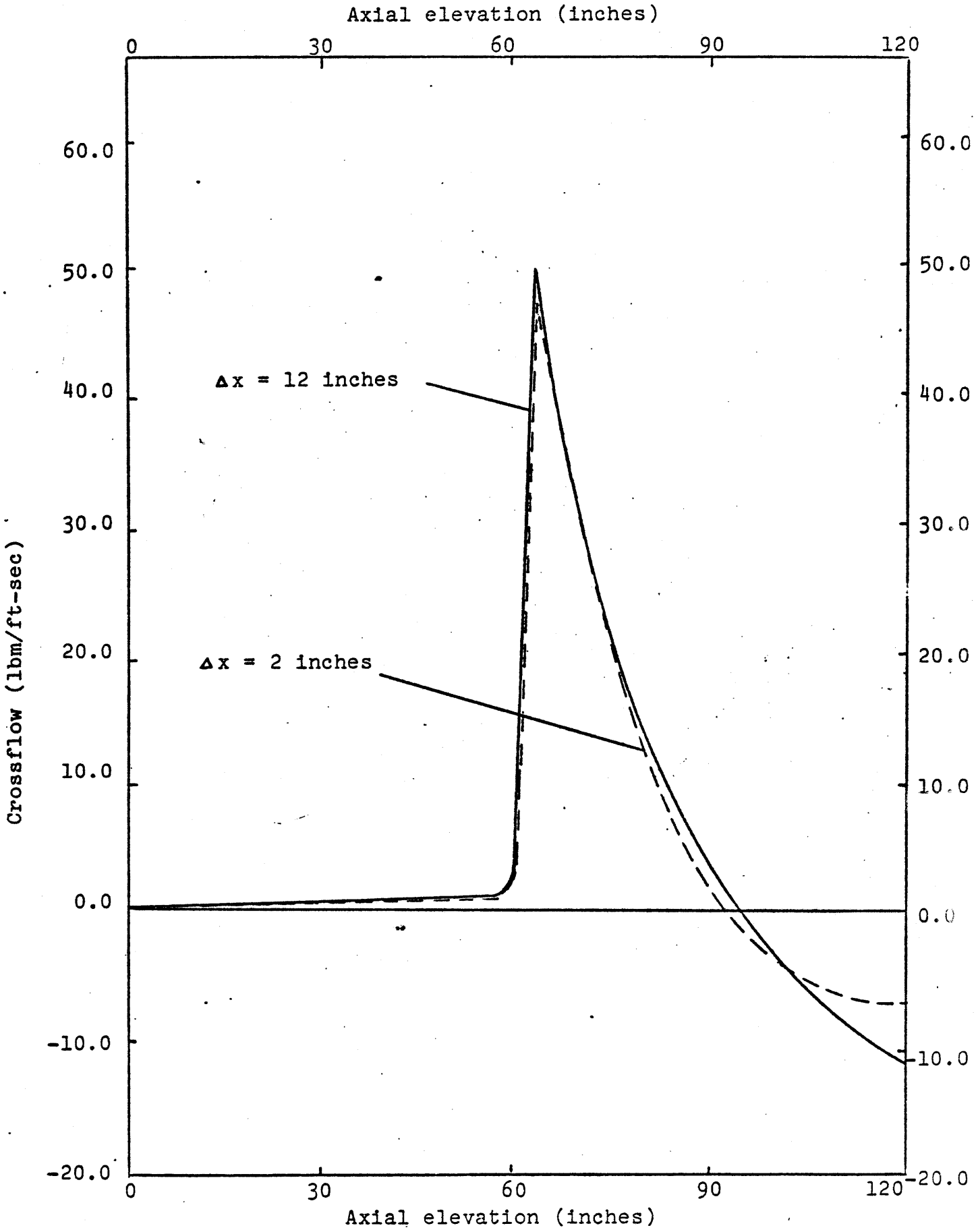


Fig. 7.19 MAT Method Blockage Predictions at t = 1 Second

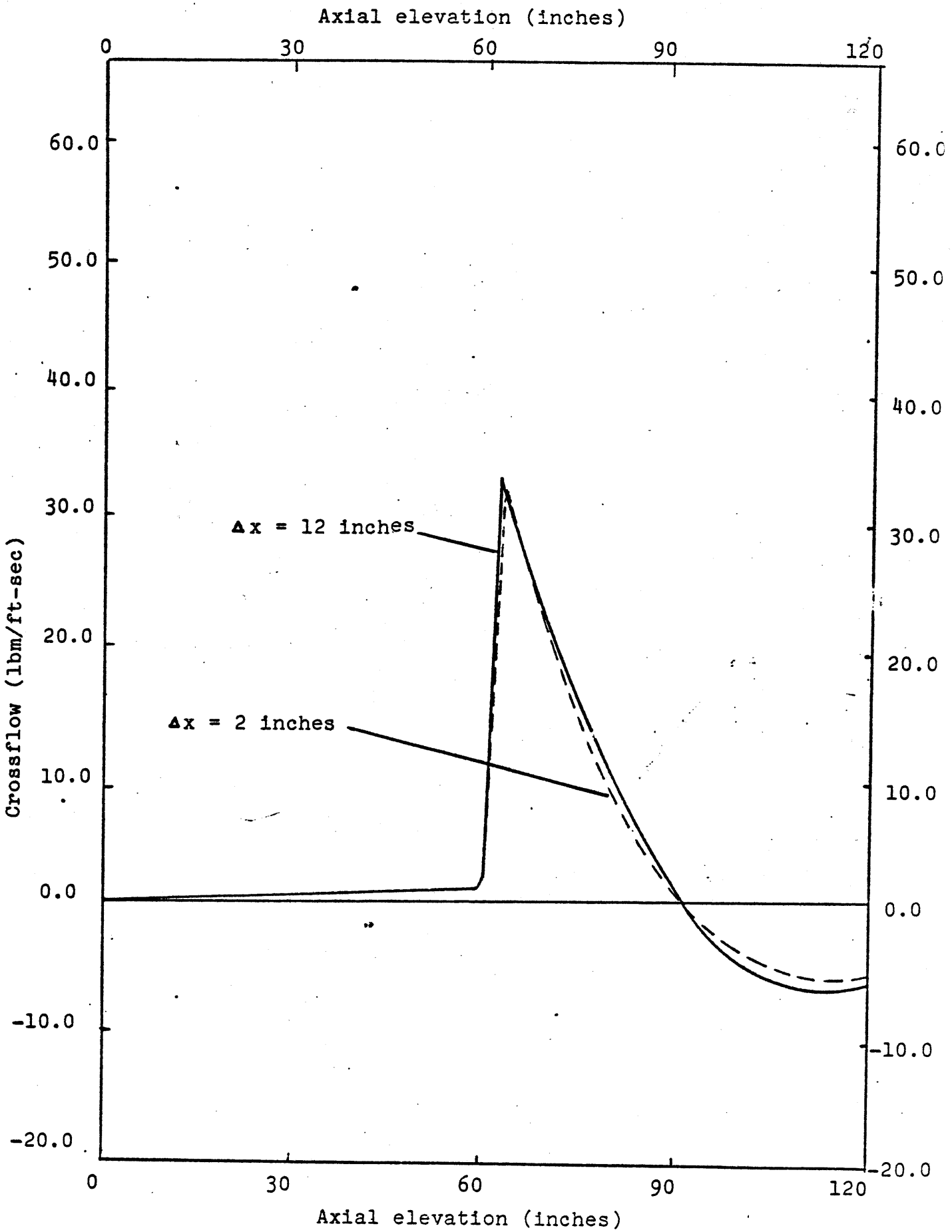


Fig. 7.20 MAT Method Blockage Predictions at t = 2 Seconds

and 7.20 show the crossflow distribution predicted by the MAT method as a function of time. It is found that there is no appreciable difference between the predictions of the method for these two different values of the flow convergence criterion, and that the results appear to approach an asymoptotic flow distribution very rapidly as additional iterations are performed or as the space step size and the time step size is reduced. The MAT method was not able to anticipate the presence of the grid at upstream locations in this case because it was implemented into the COBRA-IIIP/MIT code with a single pass, locally iterative axial iteration scheme.

The predictions of the COBRA-IIIC code are shown in Figures 7.21, 7.22, and 7.23 for a convergence criterion of .05 and in Figures 7.24, 7.25, and 7.26 for a convergence criterion of .01. It can clearly be seen in both cases that appreciable oscillations develop in the crossflow distribution in front of the blockage. Moreover, it appears that tightening the convergence criterion from .05 to .01 for this case does not necessarily make the answers any "better" if they are judged from the viewpoint of physical or numerical consistency, since there is apparently no indication that tightening the convergence criterion damps out the oscillations and causes the crossflow to converge to a truly "asymoptotic" distribution.

It must be emphasized again that it is not understood precisely why these oscillations occur. As discussed earlier,

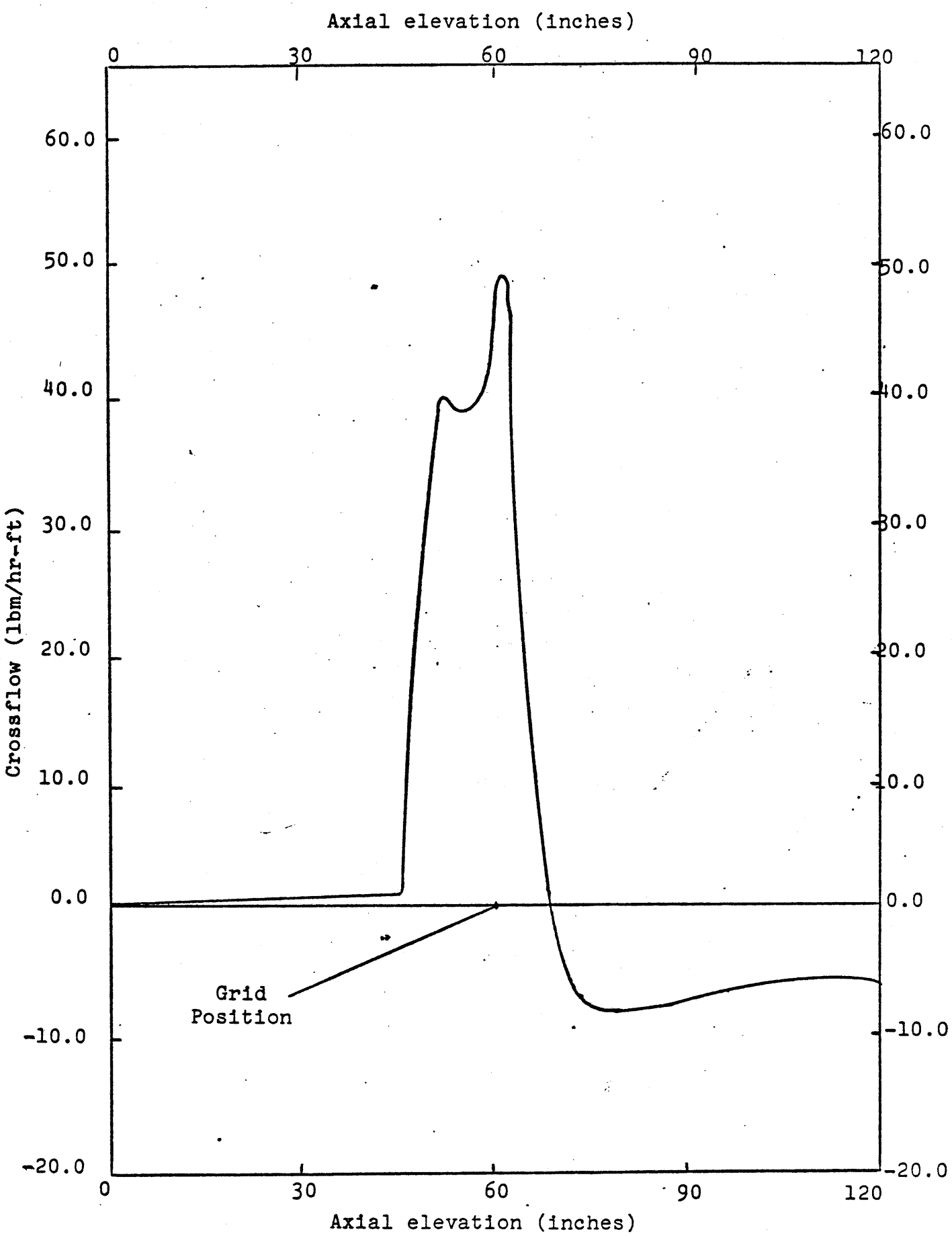


Fig. 7.21 COBRA-IIIC Blockage Predictions at t = 0 Seconds and $\varepsilon = .05$

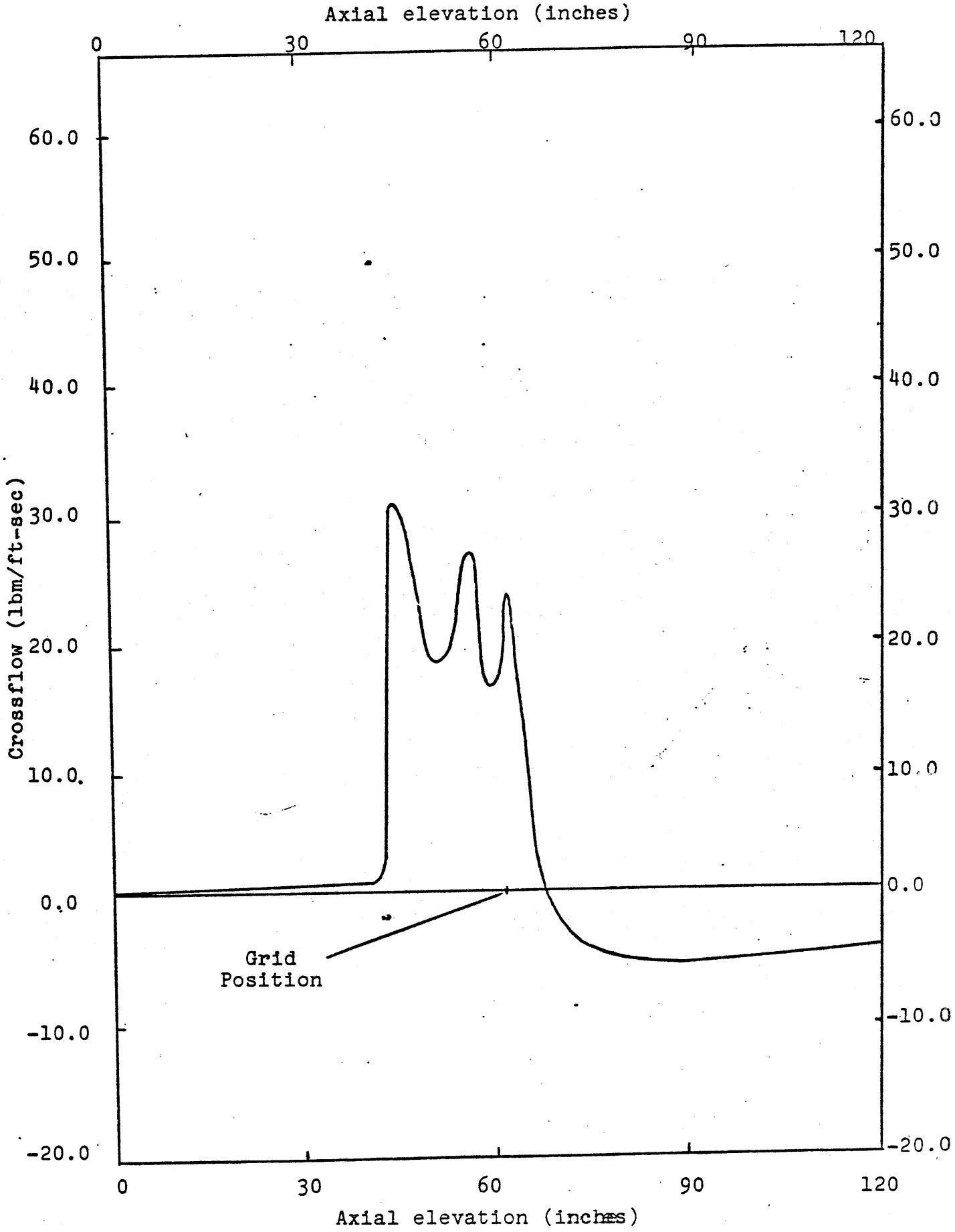


Fig. 7.22 COBRA-IIIC Blockage Predictions at t = 1 Second and ε = .05

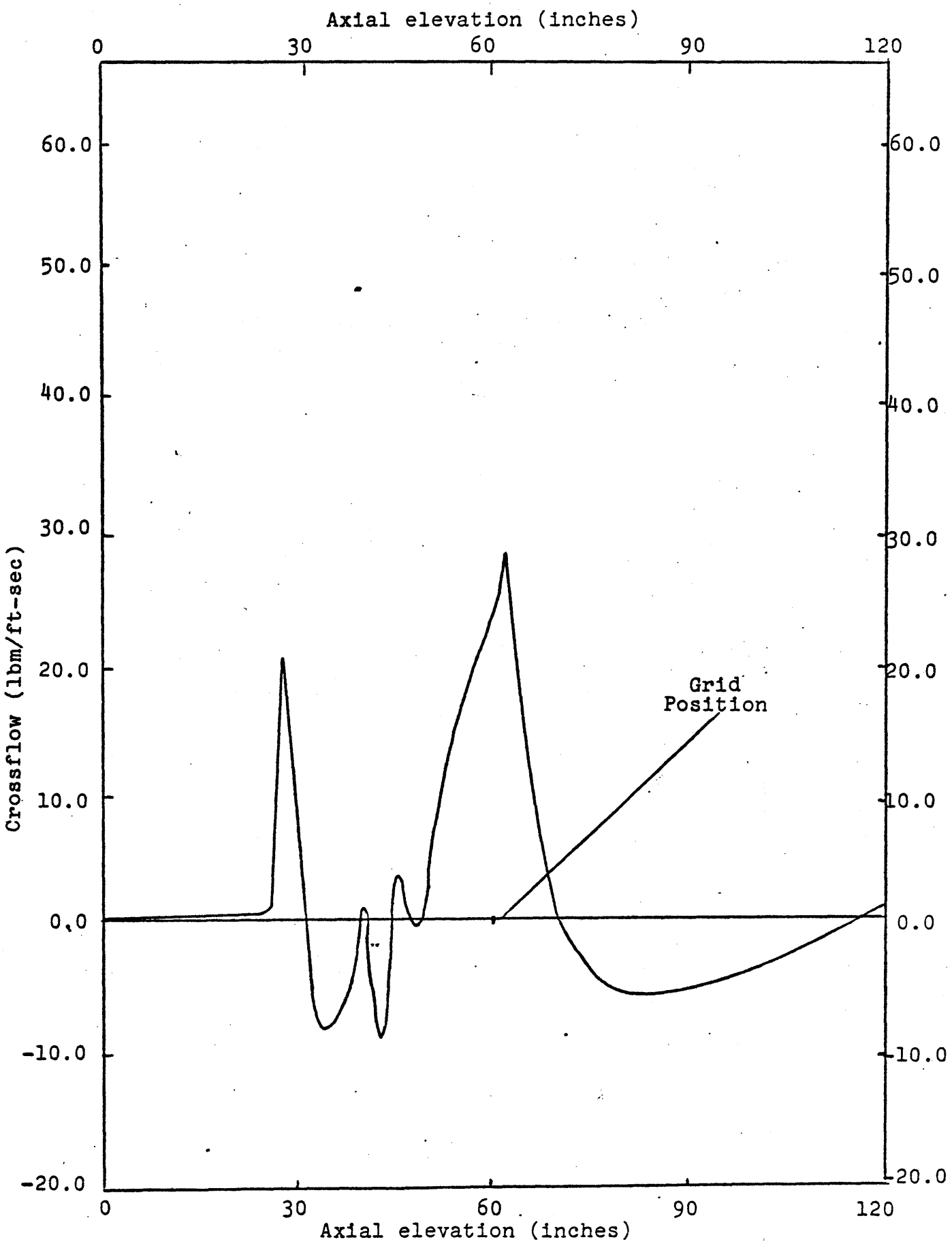


Fig 7.23 COBRA-IIIC Blockage Predictions at t = 2 Seconds and ε = .05

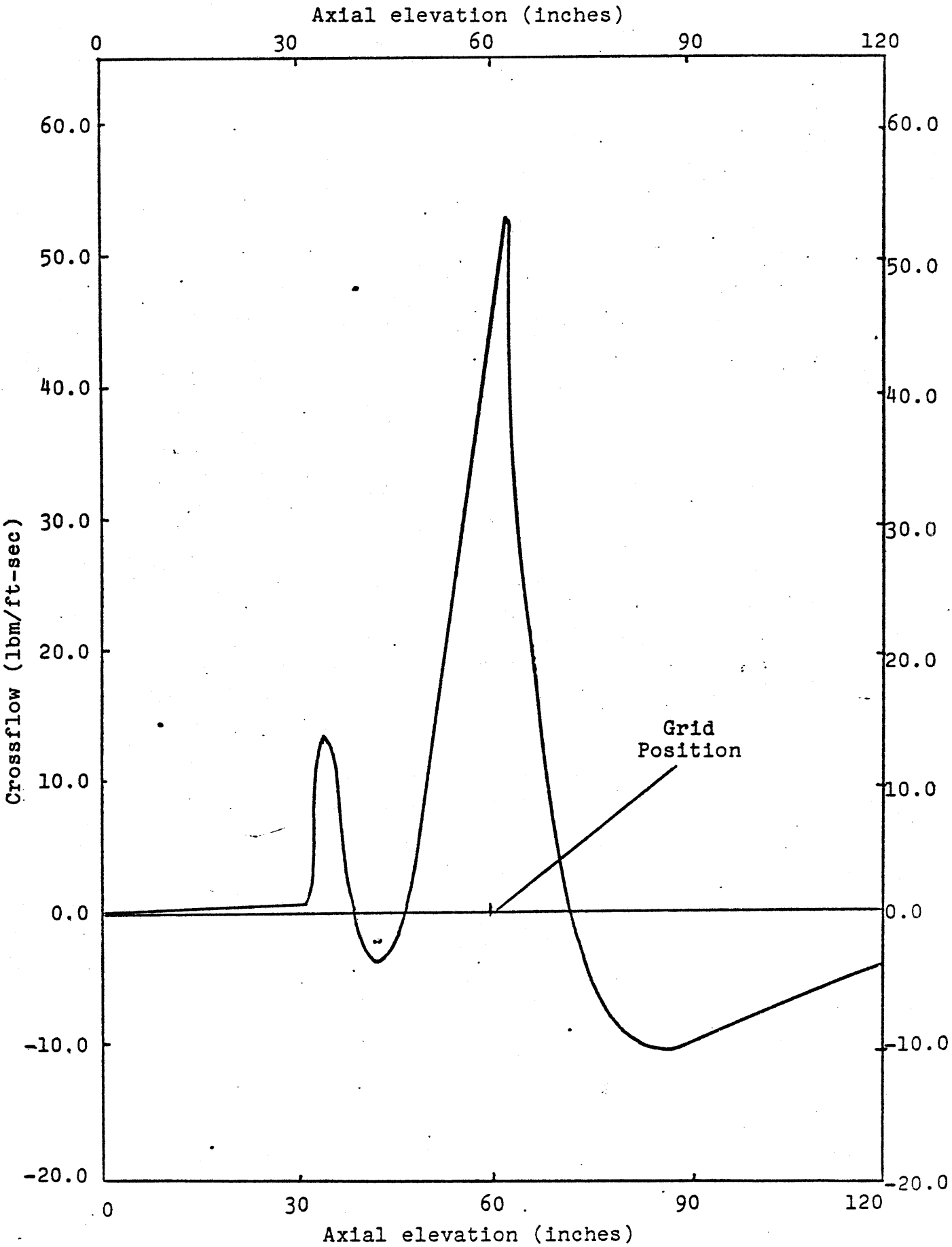


Fig. 7.24 COBRA-IIIC Blockage Predictions for t = 0 Seconds and ε = .01

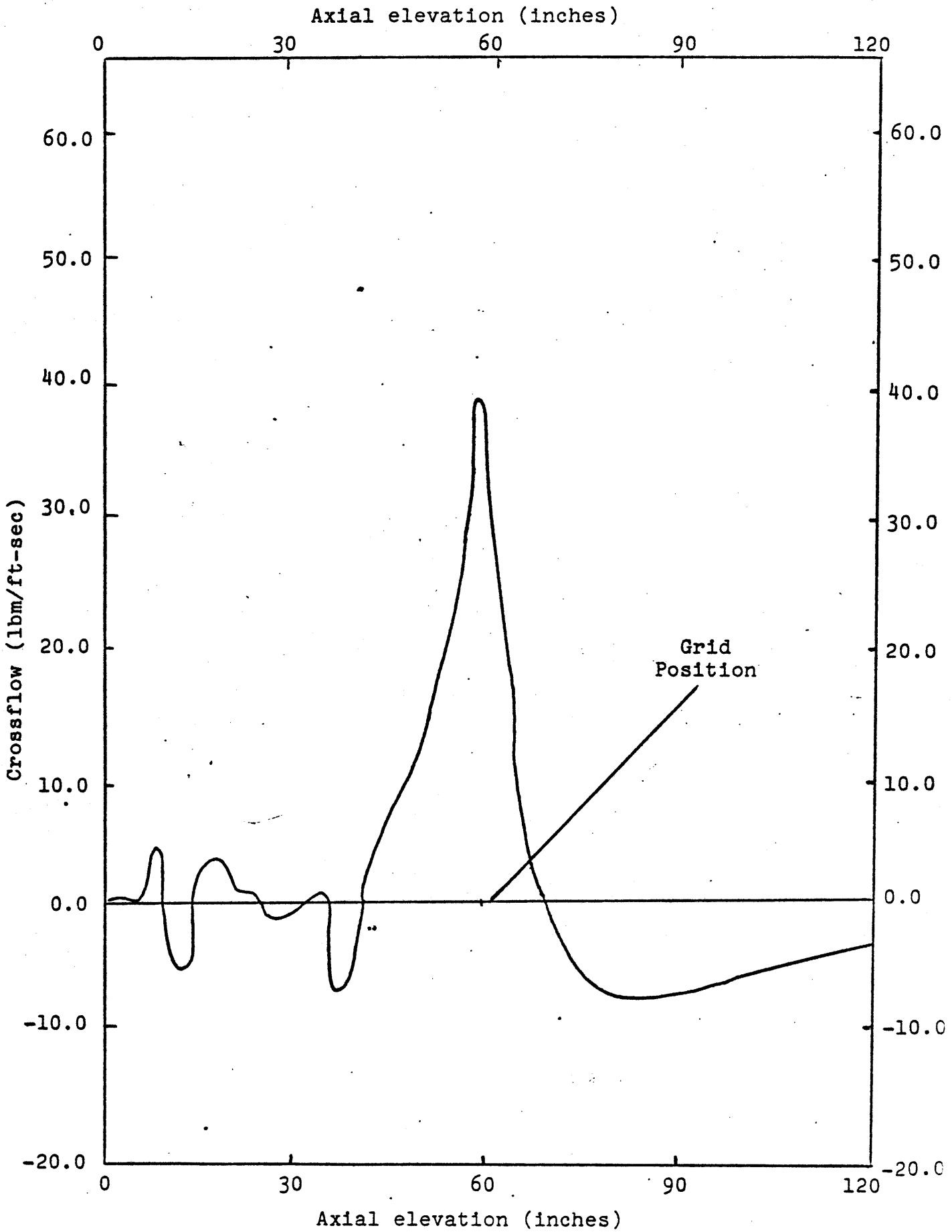


Fig. 7.25 COBRA-IIIC Blockage Predictions for t = 1 Second and $\varepsilon = .01$

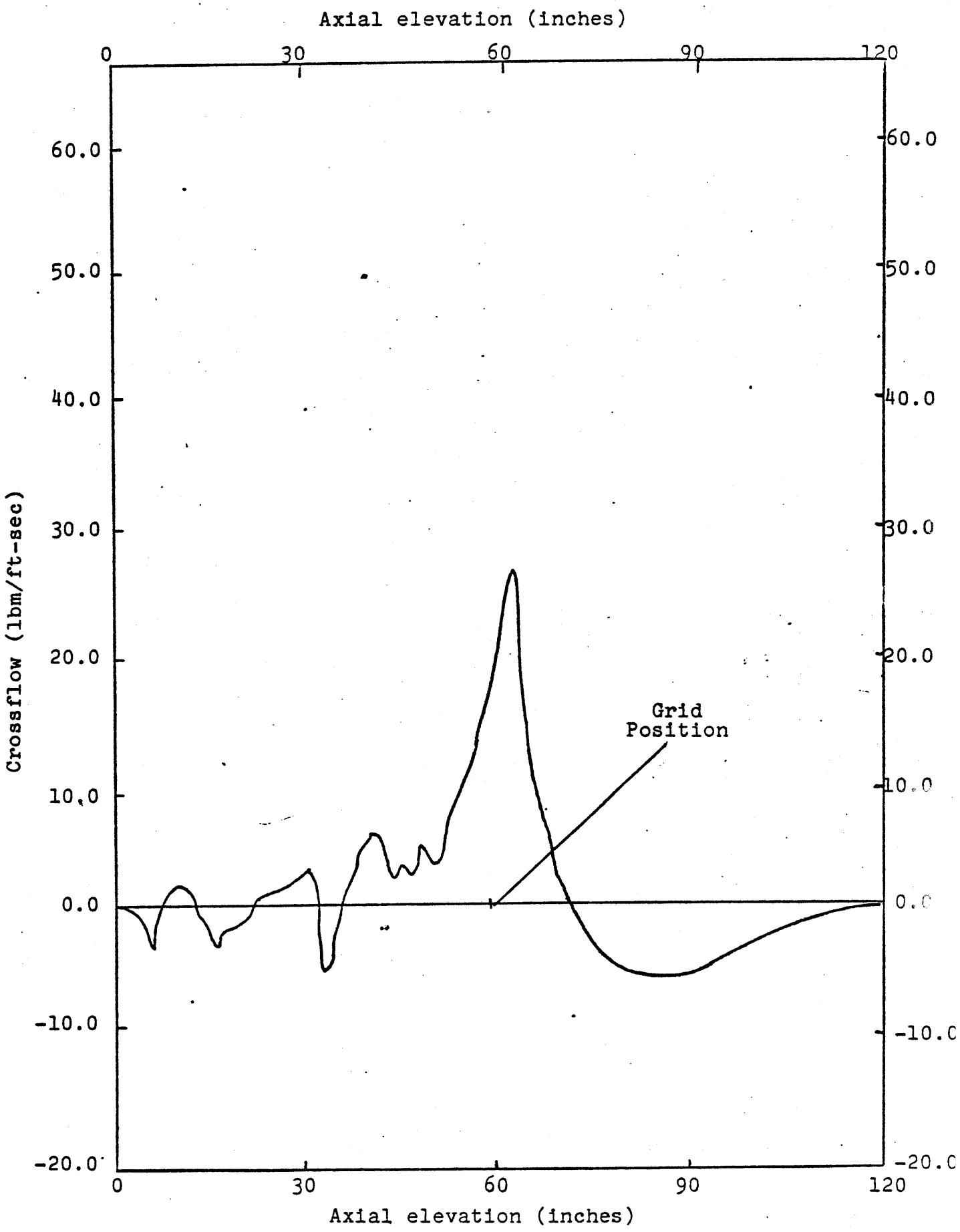


Fig. 7.26 COBRA-IIIC Blockage Predictions for t = 2 Seconds and ε = .01

it is conceivable they may be caused by roundoff errors within the code, but it is also possible that the axial iteration scheme is not converging properly, since it has never been shown conclusively that the scheme should converge to the true solution of the differential equations. It is also significant that the oscillations in this case only appear to be produced in front of the blockage (e.g. upstream), whereas in the previous case, which did not contain a blockage, they were found to be produced over the entire axial length of the bundle.

From an historical point of view, it must be pointed out that the primary reason a true multipass axial iteration scheme was put into the COBRA-IIIC code was to allow the code to propagate information from downstream locations to upstream locations at the rate of one axial node per iteration. This was necessary to provide a mechanism to enable the marching type of solution scheme used by the code to anticipate the presence of blockages, bulk boiling, and grids at positions upstream of those at which they actually occurred. It can be seen from Figs. 7.21, 7.22, and 7.23 that the code performs this function very well, but in doing so, it also appears to introduce a great deal of "fine structure" into the crossflow distribution that cannot be explained physically. It is conceivable that this "fine structure" was never reported before simply because the details of the crossflow distributions were never really analyzed by many of those who used the code.

Consequently, a great deal of attention was focused on plots of the axial mass flow rate, the coolant temperature, and the DNBR (which were not particularly sensitive to changes in the shape of the crossflow distribution) and it was then concluded that there were no anomalies caused by the axial iteration scheme simply because they were not large enough to be seen in the body of existing experimental data. From a practical point of view, it is easy to understand this approach, and it can even be argued that errors in the shape of the crossflow distribution are not important for most practical problems (as shown in Section 7.5). However, it is felt that these errors can be very important if they are large enough and appear frequently enough in the analysis of problems which contain severe flow blockages. Table 7.4 gives a comparison of the results predicted by the MAT method (with a single pass locally iterative axial iteration scheme) and the COBRA-IIIC method (with a multipass axial iteration scheme) at the outlet of the hot assembly for the flow blockage case which was just considered. It can be seen that all the variables computed by the codes agree with one another to within 2%, except for the axial mass flow rates, which can differ in some cases by as much as 10%. It is not presently known which of these results is more physically correct, but on the basis of the information that is available at this time [2,26,27] it appears that the presence of a flow blockage in an assembly is only felt by the fluid 1 or 2 inches upstream. Consequently, it is conceivable that reality may lie somewhere in between the

Table 7.4. Comparison of the Predictions of the MAT Method and the COBRA-IIIC Method at the Outlet of the Hot Channel for a Blockage Run

Time(sec)	Enthalpy(BTU/lbm)		Temperature (°F)		Density(lbm/ft^3)		Mass Flux($Mlbm/hr ft^2$)	
	COBRA-3C	MAT	COBRA-3C	MAT	COBRA-3C	MAT	COBRA-3C	MAT
0.0	632.50	640.70	610.52	615.96	41.37	40.87	2.2582	2.0654
1.0	652.95	662.53	623.87	629.87	40.10	39.49	1.7088	1.5653
2.0	696.35	708.59	642.82	642.80	32.92	29.01	1.1386	1.0239

Average enthalpy difference ≈2%

Average temperature difference ≈1%

Average density difference ≈2%

Average mass flow difference ≈ 8%

the predictions of the two codes, and that neither a single pass locally iterative scheme nor a true multipass may be able to give an entirely satisfactory answer for such an extreme case. Moreover, it is necessary to point out that if a true multipass axial iteration scheme were used with the MAT method in the COBRA-IIIP/MIT code, it would not be surprising if both the codes agreed with one another to within the limits of today's existing experimental data for blockage calculations, even if severe unphysical oscillations in the crossflow distribution were predicted by both codes in the vicinity of the blockage. Therefore, it is felt that unless experiments are performed that are able to resolve the fine structure of the crossflow distribution satisfactorily between both fuel assemblies and subchannels during transient and steady state conditions, and are able to show exactly how far upstream the presence of a blockage can be felt in an operational pressurized water reactor, the ultimate determination of the worth of the axial iteration schemes used by these codes for both a flow blockage analysis and operational reactor conditions must be measured by comparing their predictions to those of codes such as COBRA-IV-I with its MAC methodology or SABRE with its primitive variable approach and advanced models of turbulence. Only then will it be possible to answer many of the questions that have been posed in this thesis.

CHAPTER VIII

8.1 Conclusions

Based on the work that has been performed in this thesis, it is possible to draw a number of conclusions about the state-of-the-art of reactor thermal-hydraulic analysis codes, and the numerical methods that have been developed over the years to solve the fluid conservation equations used by these codes.

Briefly speaking, these conclusions can be summarized as follows:

(1) It appears that both of the numerical methods that have been developed in this thesis are considerably faster and more efficient than previous methods that have been used for solving the fluid conservation equations used by the COBRA-IIIC code.

(2) There appears to be very little difference between the predictions of these methods and those of the COBRA-IIIC code as long as they are used to analyze operational reactor conditions.

(3) For almost any practical problem, it appears to be possible to obtain good estimates of the flow and enthalpy fields with only a single pass locally iterative axial iteration scheme.

(4) Additional axial iterations do not always appear to improve the accuracy of the crossflow solution scheme in the COBRA-

IIIC code, and in many cases, they appear to make it considerably worse because they introduce severe oscillations into the crossflow distribution that cannot be explained physically.

(5) While these oscillations have been seen in many simple problems, it is not known precisely why they occur. It is possible that they are due to roundoff errors within the code, although they may also be caused by the axial iteration scheme attempting to transfer information from downstream to upstream locations.

(6) It is conceivable that these oscillations were never reported before simply because they did not have much of an effect upon the predictions of the code. It is not known whether these oscillations can be observed experimentally or if they can have a significant effect upon the results of a flow blockage analysis.

(7) From the viewpoint of computational efficiency, it apparently makes no sense to perform additional axial iterations with COBRA-IIIC if doing so consumes a great deal of additional computation time and simply makes the answers to many problems worse.

(8) The single pass locally iterative scheme used with the MAT method in the COBRA-IIIP/MIT code does <u>not</u> appear to suffer from these drawbacks and converges very rapidly to an "asymptotic" crossflow distribution without oscillations as additional iterations are performed.

(9) For problems that contain a severe flow blockage, there are perceptible differences between the predictions of the two codes, but these may be due simply to differences in the axial iteration schemes used by the codes.

(10) It is believed that the merits of all the codes and methods that have been discussed here must be judged on the basis of their computational consistency and their agreement with experimental data, since no absolute proof of the convergence or the stability of the methods used to solve the COBRA-IIIC conservation equations has ever been given.

8.2 Recommendations

Based on the observations and conclusions that have been presented here, it is believed that the predictions of the two codes are in reasonably good agreement with one another for operational reactor conditions, and that it may not be possible to resolve the small differences between the predictions of the codes experimentally for many practical problems. Consequently, it is felt that any of the codes and the methods that have been discussed here can be used with a reasonable degree of confidence for most problems that do not involve extreme departures from operational reactor conditions. Therefore, it is recommended that the COBRA-IIIP/MIT code should be used in place of the COBRA-IIIC code for dealing with these problems because of its greater speed and computational efficiency.

The results of the codes only appear to differ from each other significantly when they are used to model severe transients or problems where the effects of severe flow blockages must be taken into account. For these problems, it is recommended that an attempt should be made to determine whether the oscillations in the crossflow distribution predicted by the COBRA-IIIC code are really physical, or whether they are just numerical effects generated by the code's axial iteration scheme. If it is not possible to determine if the oscillations actually exist by comparing the predictions of the COBRA-IIIC code to those of more advanced codes such as THINC-IV and SABRE, it is recommended that the validity of the COBRA-IIIC axial iteration scheme should be tested by performing more precise experimental measurements. Although the COBRA-IIIP/MIT code has been developed mainly for use in LWR applications, it should also be recognized that the methodology developed here can be applied equally well to analyze the steady state and transient performance of LMFBR cores. This entails the application of equally efficient cell numbering schemes to deal with the hexagonal arrays encountered in these systems, and the development of a wire wrap model that is compatible with the pressure based solution schemes presented here. Finally, it is recommended that the axial iteration scheme which is currently used with the MAT method should be converted into a true multipass scheme before the predictions of the COBRA-IIIP/MIT code are compared to the results of the flow blockage experiments that are available in the literature today. Only then will it be possible to answer many of the questions that have been posed in this thesis.

REFERENCES

[1] J. Weisman and R. W. Bowring, "Methods for Detailed Thermal and Hydraulic Analysis of Water-Cooled Reactors," Nucl. Sci. Eng., 57, 255 (1975).

[2] P. T. Chu, H. Chelemer, and L. E. Hochreiter, "THINC-IV, An Improved Program for Thermal-Hydraulic Analysis of Rod Bundle Cores," WCAP-7956, Westinghouse Electric Corporation (1973).

[3] B. R. Hao and J. M. Alcorn, "LYNX1-Reactor Fuel Assembly Thermal-Hydraulic Analysis Code," BAW-10129, Babcock & Wilcox (1976).

[4] B. R. Hopper and J. R. Gloudemans," LYNX2-Subchannel Thermal-Hydraulic Analysis Program," BAW 10130, Babcock & Wilcox (1976).

[5] K. G. Becker and P. Persson, "Predictions of Burnout Conditions for Full-Scale 36 Rod Bundles with Uniform and Non-Uniform Heat Flux Employing the COBRA Subchannel Analysis and the Becker Burnout Correlations," Paper D-2, European Two-Phase Flow Meeting, Milan, Italy (June 1970).

[6] R. W. Bowring, "HAMBO, A Computer Programme for the Subchannel Analysis of the Hydraulic and Burnout Characteristics of Rod Clusters, Part 1, General Description," AEEW-R524. Atomic Energy Establishment, Winfrith, England (1967).

[7] D. S. Rowe, "COBRA-IIIC: A Digital Computer Program for the Steady State and Transient Thermal-Hydraulic Analysis of Rod Bundle Nuclear Fuel Elements," BNWL-B-82, 1971.

[8] D. S. Rowe, "COBRA-II: A Digital Computer Program for Thermal-Hydraulic Subchannel Analysis of Rod Bundle Nuclear Fuel Elements," BNWL-1229, Battelle Pacific Northwest Laboratories (February, 1970).

[9] R. Bowring and P. Moreno, "COBRA-IIIC/MIT Computer Code Manual," Dept. of Nuclear Engineering, M. I. T., (March 1976), unpublished.

[10] C. L. Wheeler, et al., "COBRA-IV-I: An Interim Version of COBRA for Transient Thermal-Hydraulic Analysis of Rod Bundle Nuclear Fuel Elements and Cores," BNWL-1962, Battelle Pacific Northwest Laboratories (March 1976).

[11] C. W. Stewart and D. S. Rowe, "Advanced Continuous Fluid Eulerian Computation Scheme for Flows with Large Density Gradients," Trans. Am. Nucl. Soc. 24, 178 (1976).

[12] J. E. Welch, et al., "The MAC Method, A Computing Technique for Solving Viscous, Incompressible, Transient Fluid-Flow Problems Involving Free Surfaces." LA-3425, (November 1965).

[13] A. D. Grossman, et al., "The SABRE Code for Prediction of Coolant Flows and Temperatures in Pin Bundles Containing Blockages," AEEW-R905, U. K. Atomic Energy Authority, Winfrith (1973).

[14] Proprietary Computer Program of Westinghouse Electric Co., Pittsburgh, Pa.

[15] Proprietary Computer Program of General Electric Company, San Jose, California.

[16] R. E. Masterson and L. Wolf, "An Efficient Multi-dimensional Numerical Method for the Thermal-Hydraulic Analysis of Nuclear Reactor Cores," ANS Topical Meeting, Tuscon, Arizona, March 1977 to be published in Nucl.Sci.and Eng. (Summer, 1977).

[17] Massachusetts Institute of Technology," MEKIN: MIT-EPRI Nuclear Reactor Core Kinetics Code," EPRI-Research Project 227 (September 1975).

[18] A. Cook, Ph.D. Thesis, Dept. of Nuclear Engineering, M. I. T., to be published.

[19] L. A. Hageman, "The Estimation of Acceleration Parameters for the Chebychev Polynomial and the Successive Over-relaxation Iteration Methods, WAPD-TM-1038, (June 1972).

[20] M. Clark and K. F. Hansen, Numerical Methods of Reactor Analysis, Academic Press, New York (1964).

[21] P. Roache, Computational Fluid Dynamics, Hermosa Publishers, Alberquerque, N. M. (1976).

[22] T. A. Porsching, "A Finite Difference Method for Thermally Expandable Fluid Transients," ANS Topical Meeting,Tuscon,Arizona, March, 1977 to be published in Nucl.Sci.and Eng. (Summer 1977).

[23] K. F. Hansen and E. U. Khan, "Development of a Method for Calculating Three-Dimensional Temperature and Flow Fields in a Nuclear Reactor Core during Transient and Blockage Conditions," Special Project, Department of Nuclear Engineering, M. I. T., (February 1976).

[24] T. A. Porsching, "Error Estimates for MAC-Like Approximations to the Linear Navier-Stokes Equations," Research Report No. 76-13, University of Pittsburgh (1976).

[25] R. D. Richtmyer and K. W. Morton, Difference Methods for Initial Value Problems, Interscience Publishers, J. Wiley and Sons, New York, New York (1967).

[26] W. D. Brown, "Flow Redistribution around Partially Blocked Coolant Channels in Pressurized Water Reactors," N. E. Thesis, Dept. of Nuclear Engineering, M. I. T. (June 1973).

[27] E. Weiss, "Flow Recovery in a Blocked Fuel Assembly," WARD-TH-416 (October 1969).

NOMENCLATURE

NF = Number of boundaries for cross flow between adjacent computational cells

NP = Number of points at which the pressure is to be found

Matrices:

Symbol	Dimension	Meaning	Special Properties
$[A_j]$	NPXNP	Matrix of cross sectional flow areas in the axial direction	Diagonal
$[c_j]$	NFXNF	Matrix of thermal conductivities between adjacent computational cells	Diagonal
$[C_j]$	NFXNF	Matrix of transverse friction coefficients between adjacent computational cells	Diagonal
$[D_j]$	NFXNF	Matrix used to compute the cross flow distribution from the transverse momentum equation	Diagonal
$[h_j]$	NPXNP	Matrix of enthalpies from each computational cell	Daigonal
$[\Delta h_j]$	NFXNF	Matrix of enthalpy differences between adjacent computational cells	Diagonal
$[h_j^*]$	NFXNF	Matrix of the enthalpies carried by the diversion cross flow between adjacent computational cells	Diagonal
$[Q_j]$	NPXNP	Matrix of volumetric heat generation rates	Diagonal
[S]	NPXNF	Interface or gap connection matrix used as a lateral differencing operator	Contains only entries having the values of 1, 0, and -1 not diagonal
$[S]^T$	NPXNP	Interface or gap connection matrix used as a lateral summation operator	Contains only entries having the values of 1, 0, and -1, not diagonal

Matrices (Continued)

Symbol	Dimension	Meaning	Special Properties
$[\Delta t_j]$	NFXNF	Matrix of lateral temperature differences between adjacent computation cells	Diagonal
$[u_j]$	NPXNP	Matrix containing the axial velocity of the fluid from each computational cell	Diagonal
$[u_j^*]$	NFXNF	Matrix containing the axial velocity of the fluid carried by the diversion crossflow between adjacent computational cells.	Diagonal

Vectors:

$\{a_j'\}$	NPX1	Vector containing the axial pressure gradients from each channel in the lattice	--
$\{b_j\}$	NPX1	Source vector used by the theta method	--
$\{b_j'\}$	NPX1	Source vector used by the MAT method	--
$\{h_j\}$	NPX1	Vector containing the enthalpies from each channel in the lattice	--
$\{m_j\}$	NPX1	Vector containing the axial mass flow rates from each channel in the lattice	--
$\{P_j\}$	NPX1	Vector containing the pressure field at axial level j	--
$\{P_{j-1}\}$	NPX1	Vector containing the pressure field at axial level j-1	--
$\{\rho_j\}$	NPX1	Vector containing the density of the fluid in each channel in the lattice	--
$\{u_j\}$	NPX1	Vector containing the axial velocity of fluid in each channel in the lattice	--
$\{u_j^*\}$	NFX1	Vector containing the axial velocity of the fluid carried by the diversion cross flow	--
$\{w_j\}$	NFX1	Vector containing the cross flow between adjacent computational cells.	--

Miscellaneous constants:

Symbol	Dimension	Meaning	Special Properties
$\frac{s}{\ell}$	---	Transverse momentum factor	--
x	---	Axial elevation	
Δx	---	Axial mesh spacing	--
Δt	---	Time step size	--
θ	---	Weighting function used by MAT method and the theta method	
-	---	Superscript used to indicate the value of a variable from a previous time	--

DISTRIBUTION

R.J. Bariboldi
ERDA Chicago Operations Office
9800 South Cass Avenue
Argonne, IL 60439

U.S. Energy Research & Development Administration
ATTN: Acting Director, Program Analysis & Evaluation Division
Idaho Operations Office
550 2nd Street
Idaho Falls, ID 83401

H. Cullingford
U.S. Energy Research and Development Administration
Engineering Design and Component Development Branch
Division of Controlled Thermonuclear Research
Washington, DC 20545

E. Davidson
U.S. Nuclear Regulatory Commission
Division of Reactor Safety Research
Washington, DC 20555

S. Fabic
U.S. Nuclear Regulatory Commission
Division of Reactor Safety Research
Washington, DC 20555

S. Israel
U.S. Nuclear Regulatory Commission
Division of Technical Review
Reactor Systems Branch
Washington, DC 20555

P. Norian
U.S. Nuclear Regulatory Commission
Division of Technical Review
Core Performance Branch
Washington, DC 20555

Zoltan Rostoczy
U.S. Nuclear Regulatory Commission
Division of Technical Review
Reactor Systems Branch
Washington, DC 20555

D.F. Ross
U.S. Nuclear Regulatory Commission
Division of Technical Review
Core Porformance Branch
Washington, DC 20555

Union Carbide Corp
Oak Ridge National Laboratory
P.O. Box Y
Oak Ridge, Tennessee 37830

M. Fontana
J. Wantland

J. Weisman
University of Cincinnati
Cincinnati, OH 45221

A.A. Bishop
Department of Chemical Engr.
University of Pittsburgh
Pittsburgh, PA 15213

H.S. Isbin
Department of Chemical Engr.
University of Minnesota
Minneapolis, MN 55455

J.E. Casterline
Department of Chemical Engr.
Columbia University
New York, NY 10027

C. Morgan
Babcock & Wilcox
P.O. Box 1260
Lynchburg, VA 24505

P. Lorenzfui
Bettis Atomic Power Laboratory
P.O. Box 79
West Mifflin, PA 15122

D. diLauro
Combustion Engineering, Inc.
Windsor, CT 06095

D.S. Rowe
Exxon Nuclear Corporation
2110 Horn Rapids Road
Richland, WA 99352

G.A. Sofer
Exxon Nuclear Corporation
2110 Horn Rapids Road
Richland, WA 99352

D. Trent
Manager, Fluid and Thermal
Engineering Division
Battelle Pacific Northwest Laboratories
Battelle Boulevard
Richland, WA 99352

G.H. Halsey
Knolls Atomic Power Laboratory
General Electric Company
P.O. Box 1072
Schenectady, NY 12301

R.A. Markley
Westinghouse Electric Corporation
Waltz Mill Site
P.O. Box 158
Madison, PA 15663

A.E. Bergles
Department of Mechanical Engr.
Iowa State University
Ames, IA 50011

J. Chen
Department of Mechanical Engr.
Lehigh University
Bethlehem, PA 18015

W. Reed
Group Leader G6 or G7
Los Alamos Scientific Laboratory
Los Alamos, NM 87545

P. Griffith
Department of Mechanical Engr.
Massachusetts Institute of Technology
Cambridge, MA 02139

S.G. Bankoff
Department of Chemical Engineering
Northwestern University
Evanston, IL 60201

L. Shotkin
U.S. Nuclear Regulatory Commission
Division of Reactor Safety Research
Washington, DC 20555

L.S. Tong
U.S. Nuclear Regulatory Commission
Division of Reactor Safety Research
Washington, DC 20555

Aerojet Nuclear Company
ATTN: Mgr. Systems Safety
Research Division
550 2nd Street
Idaho Falls, ID 83401

R. Henry
Argonne National Laboratory
9700 South Cass Avenue
Argonne, IL 60439

W. Wulff
Brookhaven National Laboratory
Upton, Long Island, NY 11973

Consolidated Edison Co. of
New York, Inc.
ATTN: Senior Vice President
4 Irving Place
New York, NY 10003

T. Fernandez
Electric Power Research Institute
3412 Hillview Avenue
P.O. Box 10412
Palo Alto, CA 94304

General Electric Company
Attn: Mgr, Reactor Systems
Safety Development
San Jose, CA 95114

Dr. R.T. Lahey
Rensselaer Polytechnic Institute
Troy, New York 12181

Robert Harris
Northeast Utilities
P.O. Box 270
Hartford, CT 06101

J. Turnage
Yankee Atomic
20 Turnpike Road
Westboro, MA 01581

J. Gosnell
Boston Edison
Nuclear Engineering Dept.
800 Boylston St.
Boston, MA 02199

R. Neuhold
Division of Reactor Development
and Demonstration
U.S. ERDA
Washington, DC 20555

E. Feldman
B uilding 362
Argonne National Laboratory
9700 South Cass Avenue
Argonne, IL 60439

J. Hansen
Hanford Engineering Development Lab.
P.O. Box 1970
Richland, WA 99352

V.J. Esposito
Manager, Safeguards Analysis I
Westinghouse Electric Corporation
P.O. Box 355
Pittsburgh, PA 15230

B. Chen
Commonwealth Edison Company
Nuclear Fuel Services
P.O. Box 767
Chicago, Illinois 60690

Dr. Ehsan Khan
Bldg. 208, Rm. E-215
Div. of Applied Physics
Argonne National Laboratory
9700 South Cass Ave.
Argonne, IL 60439

Dr. Robert Bowring
Atomic Energy Estab.
Winfrith
Dorchester
Dorset DT2 8DH
ENGLAND