Spring 5-31-1971

A new technique for steady state and transient analyses of incompressible flow networks

George V. Catanzaro
New Jersey Institute of Technology

A NEW TECHNIQUE FOR STEADY STATE AND TRANSIENT ANALYSES OF INCOMPRESSIBLE FLOW NETWORKS

BY

GEORGE V. CATANZARO

A THESIS

PRESENTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE

OF

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

AT

NEWARK COLLEGE OF ENGINEERING

Newark, New Jersey
1971

ABSTRACT

A new technique for the calculation of the transient or the steady-state mass flow rate and pressure distribution in incompressible flow networks is presented. Employing the matrix method of network analysis, the nodal continuity and branch momentum equations are solved simultaneously to obtain explicit relations giving the unknown nodal pressures and branch mass flow rates. In this manner, the transient or the steady-state behavior of incompressible flow networks with arbitrary configuration having nodal sources and sinks as well as branch transducers can be determined. In contrast with the conventional steady-state network analysis methods, the new technique can be extended to the unsteady analysis of compressible flow in networks having an arbitrary configuration with heat transfer and phase change. To ascertain the accuracy of the solution, a numerical stability and convergence analysis is performed which provides an estimate for the upper bound of the time increment needed for a stable and convergent solution. The new technique can be applied to the treatment of transient problems such as flow coastdown studies resulting from loss of pumping power in nuclear water reactors, hydraulic transients of the cooling system for large steam power plants as well as the steady-state analysis of water distribution networks. The latter application is demonstrated in this study.

APPROVAL OF THESIS

A NEW TECHNIQUE FOR STEADY STATE AND

TRANSIENT ANALYSES OF INCOMPRESSIBLE FLOW NETWORKS

BY

GEORGE V. CATANZARO

FOR

DEPARTMENT OF MECHANICAL ENGINEERING

NEWARK COLLEGE OF ENGINEERING

BY

FACULTY COMMITTEE

APPROVED:

.

.

NEWARK, NEW JERSEY

1971

ACKNOWLEDGMENTS

The author wishes to express his sincere appreciation to Dr. Amir Nahavandi whose efforts gave to this thesis impetus at its inception, guidance throughout its development, and the momentum to bring it to final copy.

TABLE OF CONTENTS

TABLE OF CONTENTS (Cont'd.)

PART TWO - USER'S MANUAL

LIST OF FIGURES

LIST OF TABLES

PART ONE

ANALYTICAL FORMULATION AND RESULTS

I. INTRODUCTION

Computation of steady-state or transient mass flow rate and pressure distribution in fluid networks with given geometric configuration and dimensions is a common problem in many branches of engineering. The design of municipal water and gas distribution systems as well as the dynamic analysis of thermal systems in conventional and nuclear power plants are a few typical examples. In the order of their analytical complexity, the problem of fluid flow in networks may be divided into the following categories:

a) Steady-state, isothermal, incompressible flow
b) Transient, isothermal, incompressible flow
c) Transient, compressible flow with heat transfer
d) Transient, compressible flow with heat transfer and phase change.

The solution to the above network problems has been generally obtained on digital computers. Several difficulties have been observed in these numerical solutions:

1) When the number of network nodes and branches become large, the computer storage requirement and running time for the solution become excessive.
2) The iterative procedure, used for the solution of steady-state problems do not always converge or may at times converge very slowly.
3) The time integration used for the solution of transient

problems may become numerically unstable or non-convergent.

4) The computer programs are not sufficiently general to treat networks of any arbitrary configuration with all system components encountered in practice.

These difficulties become more severe as one moves from problems in the "a" category to "d". This situation indicates that new techniques for the solution of fluid networks are needed. To facilitate this development, the new technique may first be divised for problems in the "a" or "b" category and later extended to "c" and "d" categories.

This study presents one such new technique for the solution of isothermal, incompressible fluid flow in networks under steady-state or transient conditions. In contrast with conventional steady-state network analysis methods, this new technique can be extended to the unsteady analysis of compressible flow networks having an arbitrary configuration with heat transfer and phase change. In the present work, the matrix method of network analysis (5)* is applied to the nodal continuity and branch momentum equations. In this manner, explicit relations for the unknown nodal pressures and branch mass flow rates

* Underlined numbers in parentheses designate references on page 41

are obtained. The present formulation is set up for the solution of problems in the "b" category; i.e., the analysis of isothermal, incompressible fluid flow in networks under transient conditions. Employing the present formulation, problems in the "a" category, i.e., the analysis of isothermal, incompressible fluid flow in networks under steady-state conditions, can be readily solved by the application of the dynamic relaxation technique (17) as shown later in this study. The extension of the present formulation to problems in "c" and "d" categories require further mathematical development and is currently in progress by other investigators at the Newark College of Engineering.

II. REVIEW OF PREVIOUS WORK

The most widely used method for the solution of problems in the "a" category, i.e., the analysis of isothermal, incompressible flow in networks under steady-state conditions, is the well-known Hardy Cross iterative procedure. This procedure is commonly divided into 1) the nodal method, and 2) the loop method (3). In the nodal method, the nodal heads are continually corrected until the mass flow rates at each node is balanced. In the loop method, the mass flow rates are initially assumed for each pipe branch so that a mass balance is maintained at each junction. The head loss between any two junctions is then calculated for each possible path between those points. The mass flow rate distribution is then adjusted until the head loss between any two junctions for each possible path between those points are sufficiently close. Employing the Hardy Cross relaxation techniques, Hoag and Weinberg (9), Graves and Branscome (8), Adams (1), Dillingham (6) and Rosenhan (20) and a number of other investigators developed digital computer programs to perform hydraulic network analyses. These programs are at times hampered by numerical instability and slow rate of convergence. Various techniques for improving the convergence problem are discussed in a number of the above references. However, these techniques cannot guarantee the convergence of the numerical solution.

The second method used for the analysis of isothermal, incompressible flow in networks under steady-state conditions is the direct solution method. In this method, the continuity and momentum equations are first expressed for every network node and branch respectively. The momentum equations are then solved algebraically for the unknown branch flows in terms of the nodal pressures. Substituting these unknown mass flow rates into the continuity equations, a system of nonlinear simultaneous equations for nodal pressures are obtained. These equations are then solved by using either the Newton-Raphson method or by the direct solution of the simultaneous linearized equations. Employing the direct solutions method, Martin and Peters (12), Pitchai (18), Shamir and Howard (21), and Marlow et al (11) developed digital computer programs to perform hydraulic network analyses. These programs require a larger computer memory storage, as compared to the programs based on the Hardy Cross Method (3, 21) and are at times hampered by the divergence of the numerical solution. This latter difficulty may be overcome if a good initial guess for the solution is available (21).

A systematic approach to the analysis and synthesis of complex networks involving unsteady flow has been reported by A. Reisman (19). This study presents a generalized structural and mathematical framework to analyze or synthesize a large class of fluid systems where transient behavior is of primary interest. However, the network analysis and the computer program required for the numerical treatment of complex fluid systems have not been presented in this article. Instead, the need for the development of a computer program with sufficient flexibility to simulate fluid networks with an arbitrary configuration is emphasized.

Analog computer methods, based on nonlinear electrical resistances, have been employed by McIlroy (13), McPherson and Radziul (14), Barker (2), Kiyose (10) and many others. The merit of this approach is that, once the circuits have been prepared, the required solutions are obtained instantaneously. The disadvantages of the analog computer method are that the simulation of large networks require a large number of analog components and considerable time for setting up the machine. For these reasons, digital machines are considered more flexible than analog computers and are generally preferred in this type of study.

The foregoing survey indicates that new techniques for the solution of fluid networks are needed. Recently, a new method suitable for both transient and steady-state analyses of isothermal, incompressible fluid networks was presented by A.N. Nahavandi (16). The main distinctive features of this method are that: 1) it lends itself to a linkage with the matrix method of network analysis; and 2) the extension of the method to the unsteady analysis of compressible flow in networks having an arbitrary configuration with heat transfer and phase change, seems to be feasible. This method constitutes the foundation for the present study.

III. SCOPE AND OBJECTIVES OF THE PRESENT STUDY

The main objectives of the present study are as follows:

1) To expand the method developed by Nahavandi (16) for the transient and steady-state analyses of isothermal, incompressible flow networks and to link this procedure with the matrix method of network analysis (5) often used in the study of D.C. electrical networks.
2) To develop a digital computer program for the transient and steady-state analyses of complex hydraulic networks based on this new technique.
3) To verify the results of the present study against the results of another computer program based on the Hardy Cross method.

The present study provides the following flexibilities:

a) The hydraulic network considered may consist of any arbitrary number of nodes and branches connected according to any desired configuration. A node is defined as any point in the system at which either three or more flows meet or network geometric dimensions change. A branch is defined as the line connecting any two nodes.
b) Any node may be connected to a number of other nodes.
c) Sources and sinks with given input and output mass flow rates may be introduced at any node.

d) Every branch may include a transducer, such as a pump or a turbine, whose differential head versus flow characteristic is known.

The present analysis is based on the following simplifying assumptions:

a) The fluid flow in the system is considered incompressible, isothermal and turbulent. No mass storage is allowed at any node or within any branch. Furthermore, the wave propagation effect, heat transfer and phase change are not considered in the analysis. In other words, this study presents a procedure for the solution of problems in categories "a" and "b" discussed in the introduction.

b) The friction factor for head loss calculations is based on correlations developed under steady-state conditions. In other words, the frequency-dependence of friction is ignored in the present analysis. Two types of frictional head loss correlations are employed: 1) Darcy-Weisbach frictional head loss equation with curve-fitted Moody's friction factor correlation (4, 15); and 2) Hazen-Williams frictional head loss equation (9, 21).

The present analysis can be applied to the treatment of many engineering problems. A few typical examples of such applications follows:

In the area of transient analysis, the present study may be applied to the solution of: 1) flow coast-down problems resulting from loss of pumping power in nuclear water reactors (7); and 2) hydraulic transients of the cooling system for large steam power plants (19).

In the area of steady-state analysis, the present study may be applied to the solution of water distribution networks. To understand this particular application, it should be borne in mind that the hydraulic transient solution can be made to approach asymptotically to the steady-state solution if the system boundary conditions are kept unchanged during the dynamic study. Thus, to perform a steady-state analysis of a water distribution network, the initial flow distribution is assumed. The system source and sink pressures and mass flow rates are then held constant during the transient analysis. Under these conditions, the nodal pressures and branch mass flow rates approach asymptotically to steady-state values. This particular application is commonly known as dynamic relaxation (17) and is further discussed in this report.

IV. MATHEMATICAL FORMULATION

In the present analysis, first the pressure drop versus pressure, nodal continuity, and branch momentum equations are expressed in matrix form. These equations are then solved to obtain explicit relations for unknown nodal pressures and branch mass flow rates as follows. An illustrative example for this procedure is presented in the Appendix.

The branch pressure drop array Δp, designating the pressure drop in the positive flow direction, is related to nodal pressures p by

$$\{\Delta p\} = [A] \{p\} \tag{1}$$

where A is the connection matrix defined as follows: Every branch of the network, under investigation, is first given an arbitrary positive orientation or direction (see Fig. 1). The connection matrix for branches and nodes is then defined as a matrix whose elements are +1, -1, or 0. only. The rows of the connection matrix correspond to the network branches such that the maximum number of rows are equal to the maximum number of system branches. The columns of the connection matrix correspond to the network nodes such that the maximum number of columns are equal to the maximum number of system nodes. If the branch orientation is "away" from the node, the corresponding element of the connection matrix is +1. If the branch orientation is "toward" the node, the corresponding element of the connection matrix is -1. If the branch and the node do not meet, the corresponding element of the connection matrix is 0.

The nodal continuity equation, considering both the branch flow and the nodal source and sink (input and output) flows is given by the following matrix equation:

$$\left[A^T\right] \left\{\dot{m}\right\} + \left\{\dot{m}_{io}\right\} = \left\{0.\right\} \tag{2}$$

where A^T is the transpose of the connection matrix and $\dot{m}$ and $\dot{m}_{io}$ are the branch flow and the nodal source and sink (input and output) flow arrays respectively.

The branch momentum equation in finite difference form in time, relating the contributions of inertia, elevational, frictional and pump head to the branch pressure drop is given by the following matrix equation:

$$\left\{\frac{\dot{m} - \dot{m}^o}{h}\right\} = \left\{\frac{ag}{L} \left(144 \;\Delta p + \rho \Delta Z + \rho H - F\right)\right\} \tag{3}$$

where

$$\left\{F\right\} = \left\{\frac{1}{2g\rho} \left(\frac{fL'}{D}\right) \frac{\dot{m}^o \;|\dot{m}^o|}{a^2}\right\} \tag{4}$$

and

$$\left\{f\right\} = \left\{.0055 \left[1.+ \left(2. \times 10^4 \;\frac{\epsilon}{D} + \frac{10.^6}{R_n}\right)^{.333}\right]\right\} \tag{5}$$

In the above relations, a, D, L and L' represent the branch flow cross-sectional area, hydraulic diameter, length and equivalent length respectively. The inertia term is based on a forward finite difference form in time as the difference between the updated and the present values of the mass flow rates ($\dot{m}$ and $\dot{m}^o$) divided by the time increment h. A criterion for the selection of this time increment will be presented later. Values of ΔZ, H, F and R_n represent the branch elevational head, pump head rise, frictional head loss and Reynolds Number; while ϵ, f and ρ designate pipe absolute roughness, fluid friction factor and density.

Equations (4) and (5) above represent Moody's method (4,15) for the calculation of frictional head loss. In order to be able to check the results of the present analysis against those of Reference (20), provisions to replace Moody's method by Hazen-Williams' method (9,21) are incorporated in the program. In the context of our present formulation, the Hazen-Williams' formulation is given by

$$\left\{ F \right\} = \left\{ \frac{3.02}{C^{1.85} \rho^{.85}} \left(\frac{L}{D^{1.165}} \right) \frac{\dot{m}^o \; |\dot{m}^o|^{.85}}{a^{1.85}} \right\} \qquad (4a)$$

The quantity C, the Hazen-Williams' coefficient, depends on the type and the condition of the conduit and is tabulated in Reference (20).

If a network consists of j branches and i nodes, equations (1), (2) and (3) constitute a set of 2j + i algebraic equations in 2j + i unknowns. These unknowns are branch pressure drops, branch mass flow rates and nodal pressures. To solve the above system of algebraic equations, we eliminate the pressure drops and mass flow rates among equations (1), (2) and (3) to obtain a single matrix equation for the unknown nodal pressures. Solving the matrix equation (3) for $\dot{m}$ array, one obtains

$$\left\{\dot{m}\right\} = \left\{\dot{m}^{o}\right\} + \left\{144 \quad \gamma\Delta p + \gamma\rho\Delta Z + \rho\gamma H - \gamma F\right\} \tag{6}$$

where

$$\left\{\gamma\right\} = \left\{\frac{agh}{L}\right\}$$

Substituting the Δp array from equation (1) into equation (6) and substituting the resulting equation into equation (2) yields

$$\left[A^{T}\right]\left\{\dot{m}^{o}\right\} + \left[A^{T}\right]\left[144\gamma\right]\left[A\right]\left\{p\right\} + \left[A^{T}\right]\left\{\rho\gamma\Delta Z + \rho\gamma H - \gamma F\right\} + \left\{\dot{m}_{io}\right\} = \left\{0.\right\} \tag{7}$$

Solving equation (7) for the unknown nodal pressure array gives

$$\left\{p\right\} = \left[M\right]^{-1}\left[A^{T}\right]\left\{-\rho\gamma\Delta Z - \rho\gamma H + \gamma F - \dot{m}^{o}\right\} - \left[M\right]^{-1}\left\{\dot{m}_{io}\right\} \tag{8}$$

where

$$\left[M\right] = \left[A^{T}\right]\left[144\gamma\right]\left[A\right] \tag{9}$$

Having the nodal pressures from equation (8), one can readily calculate the branch pressure drops and mass flow rates from equations (1) and (6) respectively. It is important to realize that $\left[144\gamma\right]$ is a diagonal matrix having branch γ's along its diagonal and zero elsewhere.

The main distinctive feature of the present analysis is that the steady-state as well as the transient nodal pressures and branch mass flow rates are calculated by the time-integration of the network transient equations as given above. Initial values of the branch mass flow rate array $\dot{m}^o$ are assumed such that continuity equations are satisfied at each node between branch flows $\dot{m}^o$ and nodal source and sink flows $\dot{m}_{io}$. Care must be taken not to confuse values of $\dot{m}^o$ and $\dot{m}_{io}$. The former quantity represents the present values of the mass flow rates which vary during the problem solution; while the latter designates fixed known flow quantities entering or leaving the system at various nodes. Following every application of equations (8), (1) and (6), the updated values of the branch mass flow rates are computed. The present values of the branch mass flow rates are then set equal to the updated values and the application of equations (8), (1) and (6) are continued. For steady-state analyses, this procedure is continued until the values of branch mass flow rates converge within a prescribed error.

To perform a transient analysis, the converged steady-state mass flow rates are entered as the initial values for $\dot{m}^o$ array. The problem solution is then restarted under the new prevailing conditions until a new set of steady-state mass flow rates is reached. This latter application can be made more clear by considering an example such as the flow coast-down problem resulting from loss of pumping power in a nuclear water reactor. Prior to the initiation of the pump coast-down transient study, the present analysis is employed once to bring the system to steady-state. The loss of pumping power is then initiated and the present analysis is employed for the second time to determine the flow behavior during the pump coast-down.

It should be noted that calculation of nodal pressures from equation (8) involves the inversion of matrix M. The elements of this matrix are independent of flow and the inversion procedure can be performed only once at the beginning of analysis. Furthermore, multiplications of M inverse by A transpose and M inverse by $\dot{m}_{io}$ are also independent of time and can also be performed once at the beginning of the analysis. The remaining operations indicated by equation (8) are time dependent and should be performed once in every time increment during the transient calculations. For large networks, the order of matrix A becomes large. This will obviously increase the computer running time for inversion, multiplications and other operations described above. This situation, however, will not create any complication in the analysis.

V. NUMERICAL SOLUTION

The numerical solution to the problem is obtained by solving equations (8), (1) and (6) on a digital computer. The input data for this analysis consists of network topography, geometric dimensions, hydraulic properties, problem initial conditions and control variables. The main input data in this study are as follows:

1) The connection matrix A (only the non-zero terms are entered as input data).
2) The branch actual length L, equivalent length L', hydraulic diameter D, nodal elevation Z arrays.
3) The fluid density ρ, viscosity μ, and pipe absolute roughness ϵ.
4) The pumps locations and their head versus volumetric flow characteristics.
5) Initial branch mass flow rates $\dot{m}^{o}$.
6) Nodal sink and source mass flow rates $\dot{m}_{io}$.
7) Integration time increment h and steady-state flow convergence error.

The numerical calculations can be conveniently divided into two groups. The first group consists of mathematical operations performed once at the beginning of the program. The assembly of the connection matrix and its transpose, the formation of matrix M defined by equation (9) and its inverse as well as the calculation of all flow-independent terms in equations (8) and (1) are among this group. The second group consists of iterative operations performed after each time increment. The calculation of branch frictional head loss and pump pressure rise as well as the computation of all flow-dependent terms in equations (8) and (1) are among this group. Classification of the numerical calculations into the above groups increases the computational efficiency of the program and reduces the computer running time.

To perform the required mathematical computations, a number of matrix operation subroutines, such as matrix addition, subtraction, multiplication, inversing and transposing were needed. These subroutines were obtained from Reference (22).

VI. NUMERICAL STABILITY ANALYSIS

The present analysis is based on the numerical integration of the branch momentum differential equations (equation 3). To obtain a convergent solution, it is necessary that the selected time increment h for this integration be smaller than the minimum "time constant" associated with these equations by a "good" margin. The momentum equations are nonlinear and an accurate calculation of their "time constants" cannot be readily made prior to their numerical solution. Fortunately, the accurate calculation of these "time constants" are not necessary and a conservative estimate is all that is needed for the purpose of obtaining a convergent solution. This objective can be achieved by making a number of simplifying assumptions. First, since the dependence of the friction factor and the pump head on mass flow rate is generally weak, it is reasonable to assume that these two variables are flow-independent. Second, the stability analysis will be performed on the linearized momentum equations. This assumption is heuristically justified because the linearized equations approximate the nonlinear equations in small time regions. If the linearized solution is convergent for every time increment, the overall nonlinear solution will also be convergent. It should be further emphasized that these simplifying assumptions are made only for the stability analysis and not for the actual analysis presented in this study.

Based on the above assumptions, the branch momentum equation may be expressed by

$$\frac{L}{ag}\frac{d\dot{m}}{dt} + K\dot{m}^2 = c \tag{10}$$

where

$$K = \frac{1}{2g\rho}\left(\frac{fL'}{D}\right)\frac{1}{a^2} \tag{11}$$

and c represents the sum of the pressure drop, the elevational head and the pump head rise. Since the linearized equation will be used only within a small time increment, we can treat both K and c as constants and find the variation of equation (10)

$$\frac{L}{ag}\frac{d(\delta\dot{m})}{dt} + 2K\dot{m}^o\,\delta\dot{m} = 0 \tag{12}$$

Equation (12) defines a linear differential equation in $\delta\dot{m}$ at each value of $\dot{m}^o$ for each time increment h. The time constant for this linear differential equation is

$$\tau_o = L \Big/ 2Kag\,\dot{m}^o \tag{13}$$

Substituting equation (11) into (13), the final expression for branch time constants become

$$\tau_o = \left(\pi D^3 \rho L\right) \Big/ 4fL'\,\dot{m}^o \tag{14}$$

Experience with the operation of the present program has indicated that if for each integration step, the time increment is smaller than the minimum time constant by a margin of 20. i.e. if:

$$h < \frac{\tau_{o\ min}}{20.} \tag{15}$$

the numerical integration is convergent. Thus, to obtain a convergent solution, the branch time constants are computed from equation (14) prior to each integration step and their minimum value is selected. The time increment for the next time interval is then chosen to be one twentieth of the minimum time constant. This margin of "20" has proved to be satisfactory for the present analysis and has resulted in a convergent solution. In the event that approximate values of the steady-state flows are known from a previous run, it is more economical to compute the time increment on the basis of these flows. The time increment can then be held constant throughout the dynamic analysis. This latter method is used in the computer program presented in this study.

The reason for using the minimum time constant can be made clear by the following argument. Any arbitrary network consists of a number of parallel and series branches. For parallel branches, the line with minimum time constant has the fastest time response and should therefore be used for numerical stability purposes. For a j number of series branches, the equivalent time constant can be easily shown to be

$$\tau = \sum_{n=1}^{j} \left(\frac{L}{ag}\right)_n \Bigg/ \left[2\dot{m}^o \sum_{n=1}^{j} (K)_n \right] \qquad (16)$$

This equivalent time constant, can be readily shown to be larger than the smallest time constant among these branches. Consequently, the minimum time constant branch provides a conservative estimate for usage in the numerical stability analysis.

When the transient solution is of no interest and only the final steady-state values are desired, the time increment criterion is not as stringent as indicated by equation (15). The time increment can be increased to the order of the system minimum time constant or even higher leading to a large reduction in computer running time. The increase in the time increment reduces the actual branch time constants without adversely affecting the steady-state results.

It should be further emphasized that numerical instability and lack of convergence constitute two major hazards of any numerical integration scheme. Numerical oscillations are caused by computational round-off errors and the lack of convergence in the result of integration truncation error. Both the numerical instability and the lack of convergence problems can be overcome by the introduction of an appropriate upper bound for the integration time step h as indicated by equation (15).

VII. PRESENTATION OF RESULTS

The new computational technique, developed in this study, is verified by employing the present digital program to determine the flow and pressure distribution in the hydraulic network shown on Figure 1. A steady-state numerical solution to this problem, based on the Hardy Cross method is available (20) and is used to check the correctness of the present dynamic analysis. The hydraulic network, shown on Figure 1, represents a typical city water distribution system. All system nodes and branches are numbered independently and positive flow directions are shown by arrows. Water enters the system through branch 1, at a rate of 2825 gpm with a constant pressure of 60 psi, and leaves the system through branches 30, 31, 32, 34 and 35 at 450, 400, 500, 650 and 825 gpm respectively. The length, internal diameter, and elevational difference for every system branch is specified in Table 1. In order to be able to match the present results against those of Reference (20), no booster pump is placed in any branch and the equivalent length of each branch is taken equal to its actual length. It should be pointed out that the present analysis is not restricted by these assumptions. These restrictions are introduced to obtain a meaningful comparison between the results of the present study and those of Reference (20).

Typical transient results are demonstrated on Figures

2 and 3. The final steady-state results are shown on Tables 2 and 3 for all system branches.

The above results are obtained by the application of the digital computer program developed for this study, designated as HYTRAN, which stands for Hydraulic Transient Analysis. The convergence criterion for reaching steady state was initially based on the relative mass flow rate error. This value was computed by subtracting the present values of the branch mass flow rates $\dot{m}^o$ from the updated values of the branch mass flow rates $\dot{m}$ and dividing the result by the updated values of the branch mass flow rates $\dot{m}$. The absolute values of these relative errors were compared with a quantity designated as steady-state flow convergence error. When the relative mass flow rate errors for all branches are smaller than the steady-state flow convergence error, steady-state conditions were considered to be reached.

Experience with the usage of this convergence criterion during the debugging of the HYTRAN program showed that this criterion is not desirable. A better criterion based on a fluid acceleration, defined by $(\dot{m}-\dot{m}^o)/h$, proved to be more satisfactory.

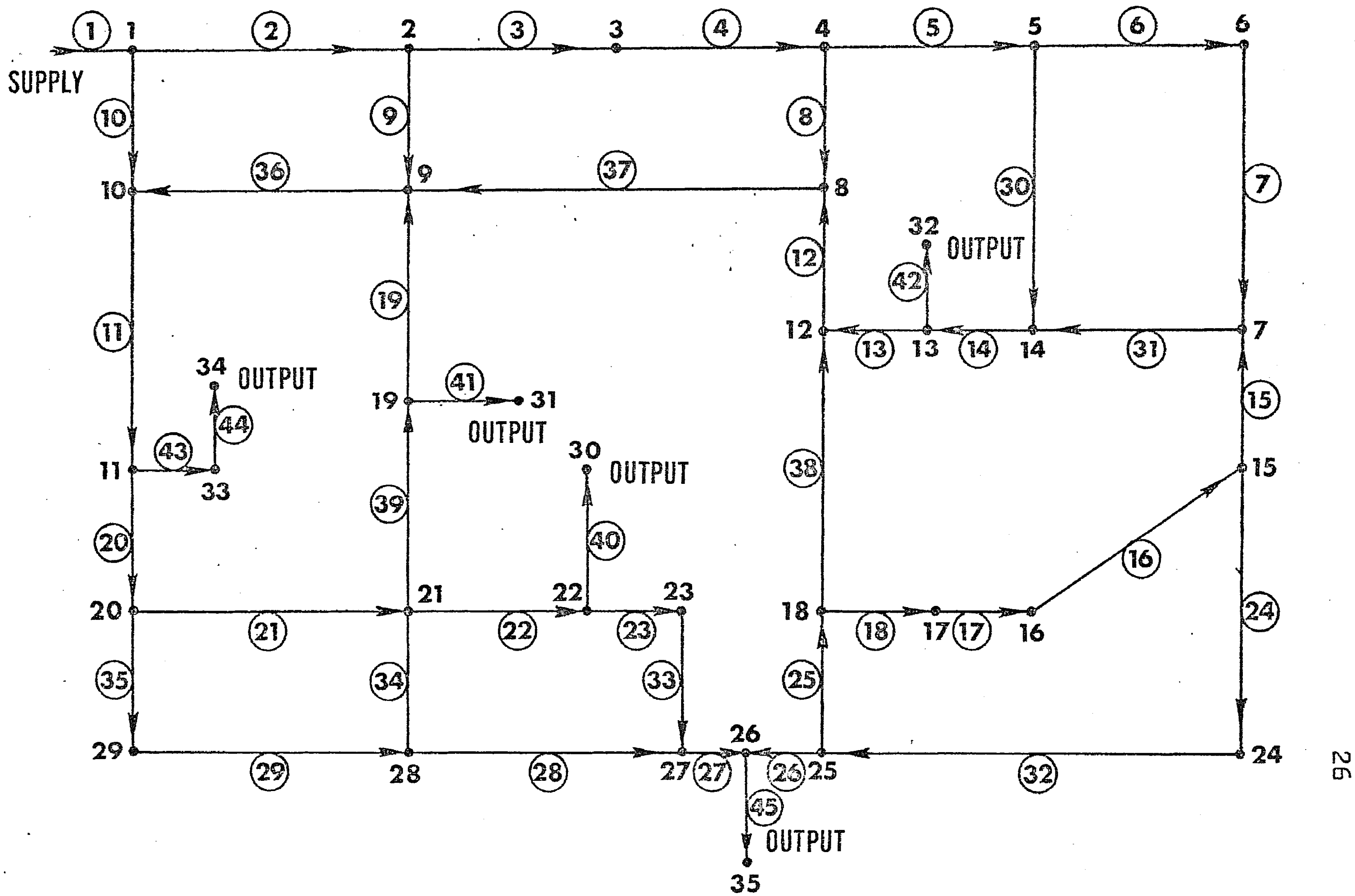


Figure 1
A Typical City Water Distribution System Employed to Verify the New Computational Technique Presented in This Study

TABLE I - GEOMETRIC SPECIFICATIONS FOR THE HYDRAULIC NETWORK SHOWN ON FIG. 1

Branch No.	Diameter D, In.	Length L, Ft.	Elevation Difference ΔZ, Ft.
1	20	1	0
2	20	200	0
3	20	150	- 10
4	20	120	+ 5
5	20	130	+ 5
6	20	100	- 10
7	10	200	- 10
8	10	50	- 5
9	10	100	- 40
10	10	100	- 40
11	8	80	- 10
12	8	50	+ 5
13	8	200	0
14	6	70	0
15	6	80	+ 10
16	5	72	0
17	5	16	0
18	5	130	- 10
19	8	120	- 10
20	6	85	+ 10
21	6	100	+ 20
22	6	70	+ 5
23	5	30	+ 5
24	5	205	- 10
25	8	90	0
26	5	20	0
27	5	30	- 5
28	5	205	0
29	5	100	+ 15
30	6	200	- 15
31	8	75	+ 5
32	5	300	+ 20
33	6	100	- 5
34	6	100	- 5
35	5	100	+ 10
36	10	200	0
37	10	300	- 30
38	5	200	+ 5
39	6	100	- 10
40	6	100	- 5
41	5	5	- 5
42	5	5	- 5
43	5	50	0
44	5	5	- 5
45	5	6	- 5

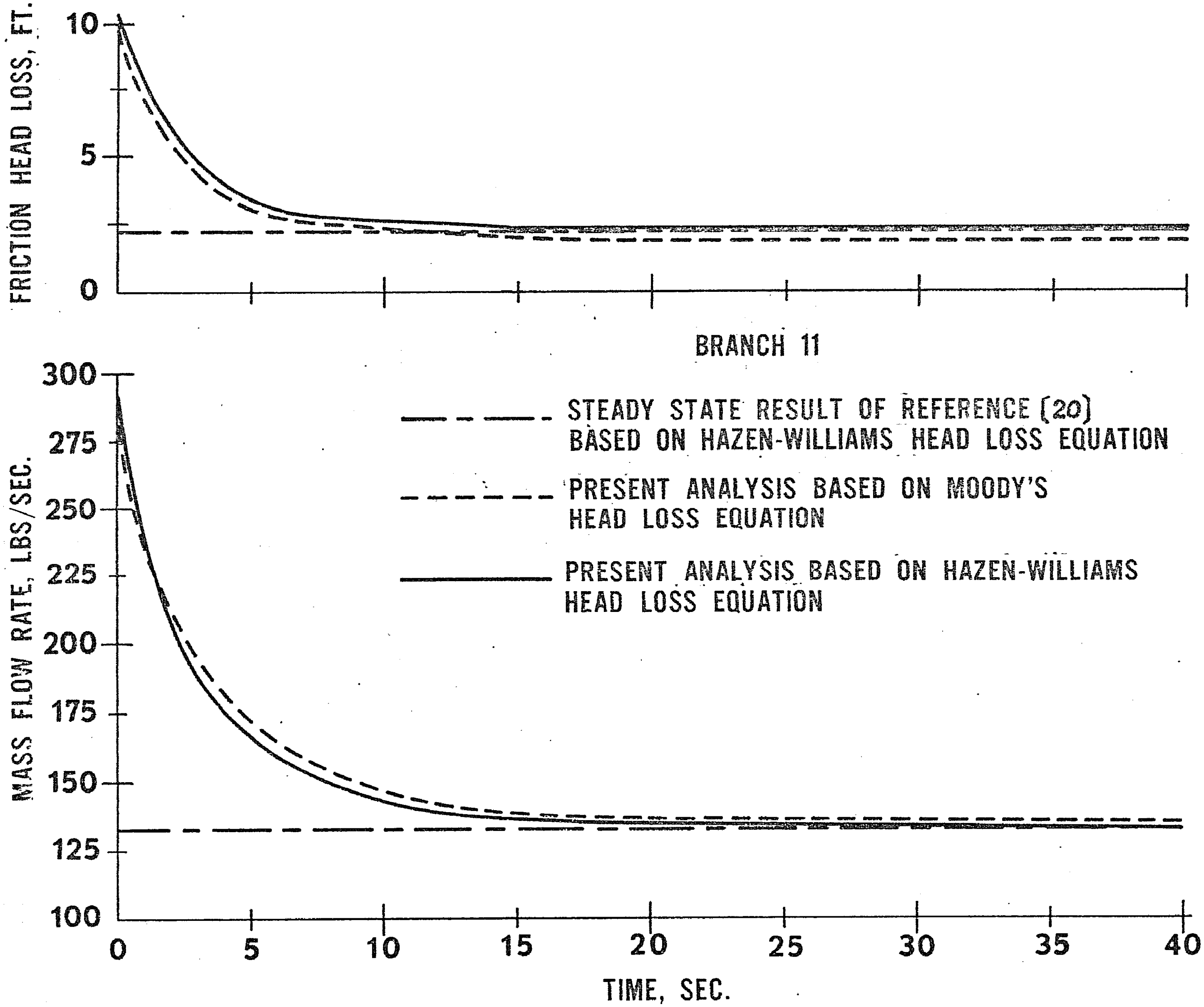


Figure 2
Typical Time Variation of Mass Flow Rate and Frictional Head Loss for Branch 11

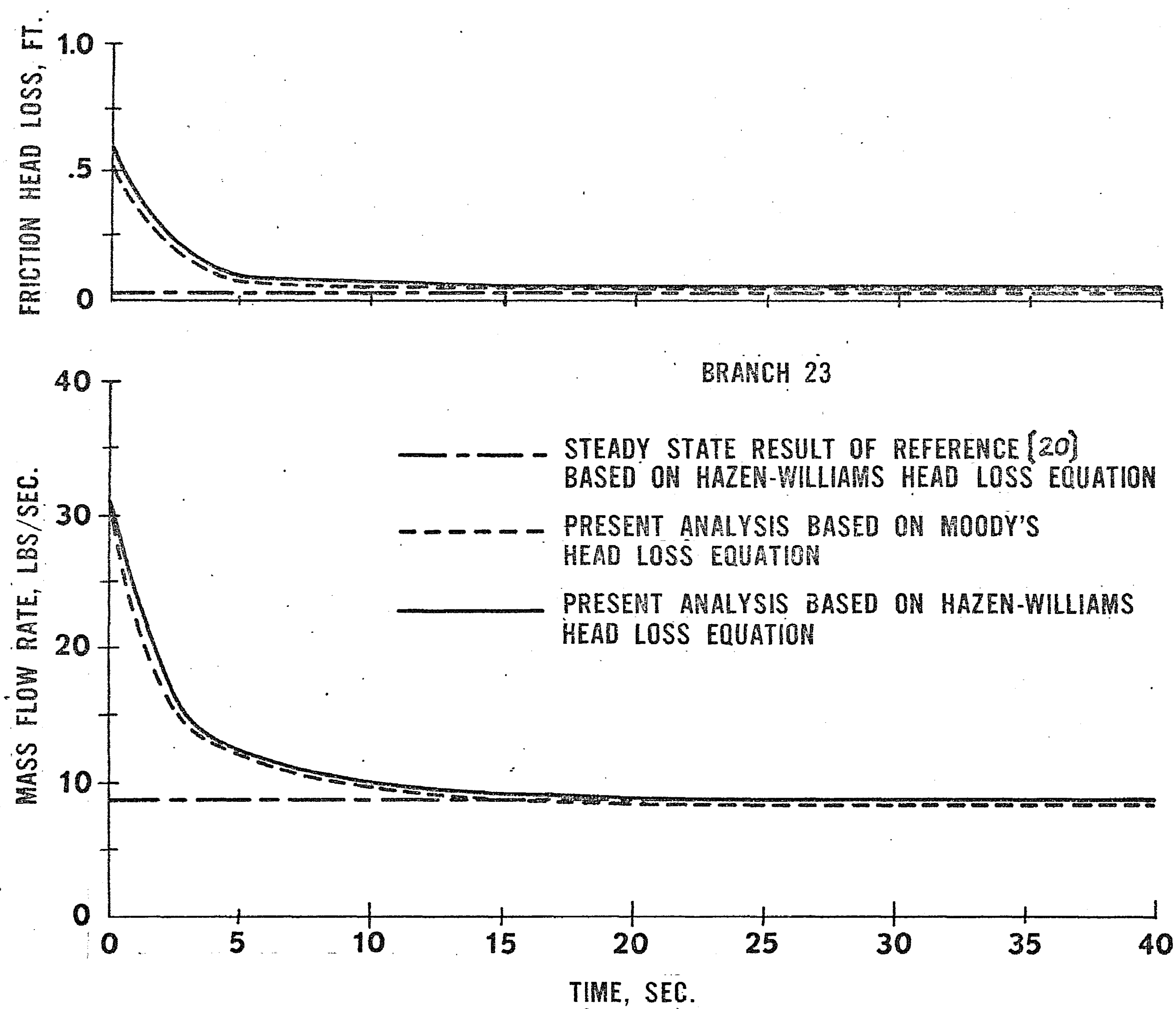


Figure 3

Typical Time Variation of Mass Flow Rate and Frictional Head Loss for Branch 23

TABLE 2 - STEADY- STATE MASS FLOW RATES (lbs./sec.) FOR THE HYDRAULIC NETWORK SHOWN ON FIG. 1

Branch No.	Present Analysis Using Moody's Head Loss Equation	Present Analysis Using Hazen-Williams' Head Loss Equation	Reference (20)
1	392.7	392.7	392.7
2	274.0	271.6	271.1
3	176.8	174.5	173.7
4	176.8	174.5	173.7
5	74.9	74.1	73.9
6	59.3	58.5	58.3
7	59.3	58.5	58.3
8	101.9	100.4	99.8
9	97.1	97.1	97.5
10	118.7	121.1	121.3
11	134.2	133.9	133.8
12	75.3	76.7	76.7
13	42.1	42.9	42.9
14	27.4	26.6	26.7
15	47.5	47.5	47.5
16	27.9	28.0	28.0
17	27.9	28.0	28.0
18	27.9	28.0	28.0
19	108.2	108.0	107.9
20	43.8	43.6	43.5
21	31.7	31.7	31.7
22	71.1	71.4	71.3
23	8.5	8.8	8.8
24	19.7	19.5	19.5
25	61.1	61.8	61.7
26	80.8	81.3	81.2
27	33.9	33.4	33.4
28	25.3	24.6	24.6
29	12.1	11.9	11.9
30	15.6	15.6	15.7
31	11.8	11.0	10.8
32	19.7	19.5	19.5
33	8.5	8.8	8.8
34	13.2	12.7	12.7
35	12.1	11.9	11.9
36	15.5	12.8	12.5
37	26.5	23.7	23.1
38	33.2	33.8	33.7
39	52.6	52.4	52.4
40	62.6	62.6	62.5
41	55.6	55.6	55.6
42	69.5	69.5	69.5
43	90.3	90.3	90.3
44	90.3	90.3	90.3
45	114.7	114.7	114.7

TABLE 3 - STEADY-STATE FRICTIONAL HEAD LOSS (ft.) FOR THE HYDRAULIC NETWORK SHOWN ON FIG. 1

Branch No.	Present Analysis Using Moody's Head Loss Equation	Present Analysis Using Hazen-Williams' Head Loss Equation	Reference (20)
1	.0	.0	.0
2	.163	.239	.23
3	.052	.079	.08
4	.042	.063	.06
5	.008	.014	.01
6	.004	.007	.01
7	.294	.409	.40
8	.213	.278	.27
9	.387	.523	.52
10	.575	.787	.77
11	1.884	2.249	2.20
12	.375	.500	.49
13	.048	.068	.07
14	.321	.402	.39
15	1.085	1.341	1.31
16	.881	1.105	1.08
17	.196	.245	.24
18	1.591	1.996	1.96
19	1.843	2.267	2.22
20	.982	1.214	1.19
21	.609	.792	.78
22	2.106	2.493	2.44
23	.036	.054	.05
24	1.264	1.604	1.57
25	.447	.605	.59
26	2.022	2.201	2.16
27	.539	.638	.63
28	2.079	2.474	2.43
29	.479	.627	.61
30	.305	.429	.43
31	.015	.021	.02
32	1.850	2.348	2.30
33	.047	.074	.07
34	.109	.146	.14
35	.239	.313	.31
36	.022	.024	.02
37	.093	.115	.11
38	3.468	4.338	4.25
39	1.655	2.012	1.97
40	2.335	2.791	2.74
41	.240	.272	.27
42	.374	.412	.40
43	6.312	6.696	6.56
44	.631	.669	.66
45	1.218	1.249	1.22

Figures 2 and 3 show the typical time-variation of mass flow rate and frictional head loss for branches 11 and 23. An examination of these curves and Tables 2 and 3 reveals the following features:

1) Application of Moody's and Hazen-Williams' frictional head loss correlations affect the shape of the system response curves. This is to be expected since the coefficients and powers appearing in equations (5) and (4a) are somewhat different. The difference between values obtained from the application of Moody and Hazen-Williams is a function of branch diameter and mass flow rate. For certain branches, the combination of diameter and mass flow rate results in a closer agreement between the response curves. The choice between Moody's or Hazen-Williams' frictional head loss correlation is a matter of personal preference. In any event, both correlations yield comparable results for practical purposes.

2) The steady-state values of branch flows and frictional head losses obtained from the present analysis, using Hazen-Williams' correlation are in a good agreement with those of Reference (20) which is also based on the same correlation. This close agreement between this analysis and Reference (20) verifies the correctness of the new technique presented in this study.

The apparent discrepancy between two frictional head losses (for branch numbers 5 and 6) is most likely due to data round-off in Reference (20).

3) Referring to the numerical stability analysis presented earlier, and noting that the system minimum time constant, based on steady-state flow, is 2 sec. (for branch 26), it can be concluded that the application of a constant time increment equal to 2.0/20. or h = .1 sec. throughout the analysis ensures a convergent solution for the problem in hand. Setting h = .1 sec., the steady-state conditions are reached after 482 iterations with a relative flow convergence error of less than .006 lb/sec^2. Decreasing the time increment to h = .05 sec. does not appreciably change the dynamic results; while increasing the time increment to h = .2 sec. affects the system response curves. This test verifies the correctness of the numerical stability analysis presented earlier in this study.

4) To provide a comparison between the computational time required by this new method against the computer program presented in Reference (20), both programs were run on an RCA Spectra 70/45 digital computer. For the steady-state analysis, since the dynamic behavior is of no interest, the time increment can be increased to the same order of the system minimum time constant

(3 sec.). This increase in the time increment reduces the actual branch time constants and speeds up the computer running time without adversely affecting the steady-state results. The computer running time required to obtain steady-state solutions for the above-mentioned hydraulic network problem is 108 sec. for the present program and 134 sec. for the program presented in Reference (20).

5) The dynamic curves shown on Figures 2 and 3 have the following physical significance. Let us assume that at time t = 0 the branch mass flow rates are those typically indicated in the Figures, and the input pressure and flow at branch 1 as well as the output flows at branches 30, 31, 32, 34 and 35 remain constant. The mass flow rate time histories in the system branches will then undergo a transient behavior typically shown in the above Figures and will approach asymptotically to final steady-state values. In other words, in the present study, both the steady-state and transient solutions are obtained by a time-dependent dynamic analysis. Values of dependent variables at intermediate time points represent the actual system behavior during a transient and are, therefore, physically meaningful. In contrast, the Hardy Cross and the direct solution methods obtain the final steady-state values after a number of iterations (9). For these

analyses, values of dependent variables at these intermediate points have no physical significance.

VIII. CONCLUSIONS

A new technique for the steady-state and transient analyses of isothermal, incompressible flow in networks is presented. In this method, the pressure drop versus pressure, nodal continuity and branch momentum equations are expressed in matrix form for networks with any arbitrary configuration. These equations are then solved to obtain explicit relations for unknown nodal pressures and branch mass flow rates.

In contrast with the conventional steady-state network analysis methods, the present technique can be extended to the transient analysis of compressible flow in networks with heat transfer and phase change. This extension is currently under development by other investigators at the Newark College of Engineering.

To ascertain the accuracy of the solution, a numerical stability and convergence analysis is performed. This analysis provides an estimate for the upper bound of the time increment needed for a stable and convergent solution.

The new technique is applied to the analysis of a water distribution network. The results obtained are in agreement with the available solution.

IX. RECOMMENDATIONS

The new technique, presented in this study, can be utilized in the development of more efficient and more economical engineering tools needed for the design of more complex fluid systems. Two such applications are described hereunder:

1) This study can be extended to analyze thermal systems involving compressible flow with heat transfer and phase change. The new program will then be able to analyze power plant and processing plant networks under steady-state and transient conditions more efficiently and more economically than is presently possible.

2) The present program can determine branch mass flow rates and nodal pressures for a system with given configuration and pipe diameters. The actual design of hydraulic networks requires the selection of pipe diameters to achieve a set of hydraulic and economic objectives. The present program can be extended to treat this design problem with improved efficiency and economy.

The present version of HYTRAN program may be improved by incorporating the following modifications:

1) A subroutine may be added to the program to compute the upper bound of the time increment automatically during

the dynamic analysis. This feature is in contrast with the presently hand-computed time increment used as an input data to the program. The procedure for the calculation of this upper bound is given in this study.

2) A laminar pressure drop versus flow relationship may be added to the program to provide a more accurate calculation of frictional losses during laminar flow regimes.

3) Expansion and contraction losses can be made dependent on the flow direction. The present version of HYTRAN assumes identical losses in both flow directions.

X. NOMENCLATURE

A	Connection matrix for branches and nodes as defined in the text
a	Branch flow cross-sectional area, ft^2
C	Hazen-Williams' coefficient
c	A constant defined in the text
D	Branch hydraulic diameter, ft
F	Pressure loss due to friction, lb/ft^2
f	Friction factor
g	Acceleration of gravity, 32.2 ft/sec^2
H	Transducer pressure differential (pump pressure rise), ft
h	Time increment, sec
K	A constant defined by equation (11)
L	Branch actual length, ft
L'	Branch equivalent length (Branch actual length plus additional lengths allowed for fittings), ft
M	A matrix defined by equation (9)
$\dot{m}$	Branch mass flow rate, lb/sec
$\dot{m}_{io}$	Nodal source and sink (input and output) mass flow rate (positive for source and negative for sink), lb/sec
p	Nodal pressure, psi
R_n	Branch Reynolds Number ($= D \dot{m}^o / a\mu$)
t	Time, sec
Z	Nodal elevations, ft

Greek Symbols

γ	A quantity defined in the text
Δp	Pressure drop along the positive flow direction, psi

$\delta\dot{m}$ A small variation of the branch mass flow rate, lb/sec

ΔZ Branch elevational difference along the positive flow direction (elevation of node at the end of the arrow minus elevation of the node at the head of the arrow), ft

ϵ Pipe roughness, ft

μ Fluid viscosity, lb/sec - ft

ρ Fluid density, lb/ft^3

τ Branch time constant, sec

Superscripts

o Present value

T Transpose of a matrix

Subscripts

Min Minimum

Miscellaneous Symbols

[ ] Contains matrices

{ } Contains arrays

XI. REFERENCES

1) Adams, R.W., "Distribution Analysis by Electronic Computer"; Journal of the Institute of Water Engineers, Vol. 15, 1961, pp. 415-428.

2) Barker, C.L., "Use of the McIlroy Analyzer on Water Distribution Systems"; Journal of the American Water Works Association, Vol. 50, January 1958, pp. 15-20.

3) Barlow, J.F., and E. Markland, "Computer Analysis of Pipe Networks"; Proceedings of the Institution of Civil Engineers, Vol. 43, May-August 1969, pp. 249-259.

4) Benedict, R.P., and N.A. Carlucci, "Flow with Losses"; Journal of Engineering for Power, Paper No. 63-WA-105, 1963, pp. 1-13.

5) Branin, Jr., F.H., "The Algebraic-Topological Basis for Network Analogies and the Vector Calculus"; IBM Technical Report TR 00.1495, July 25, 1966, pp. 1-66.

6) Dillingham, J.H., "Computer Analysis of Water Distribution Systems, Part 2"; Water and Sewage Works, Vol. 114, No. 2, February 1967, pp. 43-45.

7) Fuls, G.M., "Flow Transients Resulting from a Loss of Pumping Power in a Pressurized Water Nuclear Reactor"; ASME Publication No. 70-WA/NE-4, 1970, pp. 1-5.

8) Graves, Q.B., and D. Branscome, "Digital Computers for Pipeline Network Analysis"; Journal of the Sanitary Engineering Division, ASCE, Vol. 84, No. SA2, April 1958, pp. 1608-1 -- 1608-18.

9) Hoag, L.N., and G. Weinberg, "Pipeline Network Analysis by Electronic Digital Computer"; Journal of American Water Works Association, Vol. 49, 1957, pp. 517-524.

10) Kiyose, J., "Development of Flow Computers for Network Analysis in Japan"; Journal of the American Water Works Association, Vol. 57, October 1965, pp. 1346-1353.

11) Marlow, T.A., et al., "Improved Design of Fluid Networks with Computers"; Journal of the Hydraulic Division, ASCE, HY4, 1966, pp. 43-61.

12) Martin, D.W., and G. Peters, "The Application of Newton's Method to Network Analysis by Digital Computers"; Journal of the Institute of Water Engineers, Vol. 17, pp. 115-129.

13) McIlroy, M.S., "Direct Reading Electric Analyzer for Pipeline Network Analysis"; Journal of the American Water Works Association, Vol. 42, April 1950, p. 347.

14) McPherson, M.B., and J.V. Radziul, "Water Distribution Design and the McIlroy Network Analyzer"; Journal of the Hydraulics Division, ASCE, Vol. 84, No. HY2, 1958, pp. 1588-1 -- 1588-15.

15) Moody, L.F., "Friction Factors for Pipe Flow"; Transactions of ASME, Vol. 66, 1944, p. 671.

16) Nahavandi, A.N., "The Loss-of-Coolant Accident Analysis in Pressurized Water Reactors"; Nuclear Science and Engineering, Vol. 36, 1969, pp. 159-188.

17) Otter, J.R.H., "Dynamic Relaxation of Shell Theory Equations for Arch Dams"; Theory of Arch Dams edited by J.R. Rydzewski, Proceedings of the International Symposium held at Southampton University, England, 1964, pp. 313-318.

18) Pitchai, R., "A Model for Designing Water Distribution Pipe Networks"; Thesis presented to Harvard University, at Cambridge, Mass. in 1966 in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

19) Reisman, A., "On a Systematic Approach to the Analysis and Synthesis of Complex Systems Involving the Unsteady Flow of Fluids"; ASME Publication No. 64-WA/FE-36, 1964, pp. 1-12.

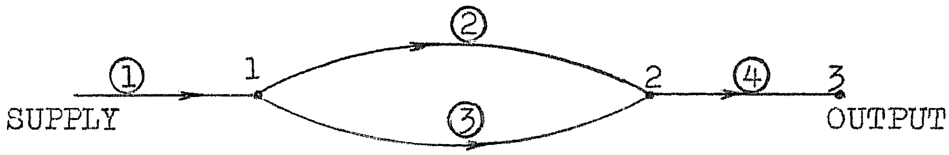
20) Rosenhan, A.K., "Computer Analysis of Flow in Pipe Networks"; ASME Publication No. 69-WA/PVP-9, 1969, pp. 1-23.

21) Shamir, U., and C.D.D. Howard, "Water Distribution Systems Analysis"; Journal of the Hydraulics Division, ASCE, Vol. 94, No. HY1, January 1968, pp. 219-234.

22) "System/360 Scientific Subroutine Package (360A-CM-03X) Version III, Programmer's Manual" International Business Machines Corporation.

XII. APPENDIX 1. ILLUSTRATIVE EXAMPLE FOR MATHEMATICAL FORMULATION

To demonstrate the mathematical formulation presented in this study, an illustrative example for a simplified network shown below is worked out in detail.



For this configuration, equation (1) takes the following form:

$$\begin{Bmatrix} \Delta p_1 \\ \Delta p_2 \\ \Delta p_3 \\ \Delta p_4 \end{Bmatrix} = \begin{bmatrix} -1 & 0 & 0 \\ +1 & -1 & 0 \\ +1 & -1 & 0 \\ 0 & +1 & -1 \end{bmatrix} \begin{Bmatrix} p_1 \\ p_2 \\ p_3 \end{Bmatrix} \qquad (17)$$

One may verify the correctness of equation (17) by a simple expansion as follows:

$$\begin{aligned} \Delta p_1 &= 0. - p_1 \\ \Delta p_2 &= p_1 - p_2 \\ \Delta p_3 &= p_1 - p_2 \\ \Delta p_4 &= p_2 - p_3 \end{aligned} \qquad (18)$$

Equations (18) are obviously the branch pressure drop versus nodal pressure relations. It should be noted that the supply pressure is considered as the reference pressure for all nodal pressures. Furthermore, equation (2) takes the following form.

$$\begin{bmatrix} -1 & +1 & +1 & 0 \\ 0 & -1 & -1 & +1 \\ 0 & 0 & 0 & -1 \end{bmatrix} \begin{Bmatrix} \dot{m}_1 \\ \dot{m}_2 \\ \dot{m}_3 \\ \dot{m}_4 \end{Bmatrix} + \begin{Bmatrix} 0 \\ 0 \\ \dot{m}_{30} \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix} \qquad (19)$$

One may verify the correctness of equation (19) by a simple expansion as follows:

$$\begin{aligned} -\dot{m}_1 + \dot{m}_2 + \dot{m}_3 &= 0 \\ -\dot{m}_2 - \dot{m}_3 + \dot{m}_4 &= 0 \\ - \dot{m}_4 + \dot{m}_{30} &= 0 \end{aligned} \qquad (20)$$

Equations (20) are obviously the nodal continuity equations for the system under consideration. It should be noted that $\dot{m}_1$, $\dot{m}_2$, $\dot{m}_3$ and $\dot{m}_4$ are the unknown branch flows; while $\dot{m}_{30}$ is the known flow output at node 3. Similarly, equation (3) takes the following form:

$$\begin{Bmatrix} \dfrac{\dot{m}_1-\dot{m}_1^o}{h} \\ \dfrac{\dot{m}_2-\dot{m}_2^o}{h} \\ \dfrac{\dot{m}_3-\dot{m}_3^o}{h} \\ \dfrac{\dot{m}_4-\dot{m}_4^o}{h} \end{Bmatrix} = \begin{Bmatrix} \dfrac{a_1 g}{L_1} & (144 \ \Delta p_1 + \rho \Delta Z_1 + \rho H_1 - F_1) \\ \dfrac{a_2 g}{L_2} & (144 \ \Delta p_2 + \rho \Delta Z_2 + \rho H_2 - F_2) \\ \dfrac{a_3 g}{L_3} & (144 \ \Delta p_3 + \rho \Delta Z_3 + \rho H_3 - F_3) \\ \dfrac{a_4 g}{L_4} & (144 \ \Delta p_4 + \rho \Delta Z_4 + \rho H_4 - F_4) \end{Bmatrix} \qquad (21)$$

It can be easily seen that equations (21) constitute the branch momentum equations. As shown earlier, solving equations (17), (19) and (21) provide explicit relations for unknown nodal pressures and branch mass flow rates. Applying equations (8) and (9) to the present case, one obtains

$$\begin{bmatrix} M \end{bmatrix} = (144gh)\begin{bmatrix} -1 & +1 & +1 & 0 \\ 0 & -1 & -1 & +1 \\ 0 & 0 & 0 & -1 \end{bmatrix}\begin{bmatrix} a_1/L_1 & 0 & 0 & 0 \\ 0 & a_2/L_2 & 0 & 0 \\ 0 & 0 & a_3/L_3 & 0 \\ 0 & 0 & 0 & a_4/L_4 \end{bmatrix}\begin{bmatrix} -1 & 0 & 0 \\ +1 & -1 & 0 \\ +1 & -1 & 0 \\ 0 & +1 & -1 \end{bmatrix} \quad (22)$$

Performing the multiplications indicated above, yields

$$\begin{bmatrix} M \end{bmatrix} = (144gh)\begin{bmatrix} (a_1/L_1)+(a_2/L_2)+(a_3/L_3) & -(a_2/L_2)-(a_3/L_3) & 0 \\ -(a_2/L_2)-(a_3/L_3) & (a_2/L_2)+(a_3/L_3)+(a_4/L_4) & -(a_4/L_4) \\ 0 & -(a_4/L_4) & (a_4/L_4) \end{bmatrix} \quad (23)$$

Inverting the above matrix, substituting into equation (8) and performing the indicated matrix operations, the nodal pressures are easily found. Subsequently, employing equations (1) and (6), the branch pressure drop and mass flow rates can be determined.

PART TWO

USER'S MANUAL

I. DESCRIPTION OF HYTRAN PROGRAM

The program HYTRAN (Hydraulic Transient Analysis) consists of a main program and several subroutine linked as shown in Figures 4 through 16. A tabular outine of the main program and its subroutines together with their functions is presented below and followed by a more detailed description.

MAIN PROGRAM	FUNCTION
MAIN	Hydraulic Transient Analysis
SUBROUTINES	**FUNCTION**
MINV	Matrix Inversion
GMPRD	Matrix Multiplication
GMSUB	Matrix Subtraction
GMTRA	Matrix Transposition
SI FUNCTION	Linear Interpolation
DPRINT	Matrix Printout

Main Program

A detailed description of the MAIN program is presented in the flow charts on Figures 4 through 15. This program is employed to perform the following two main functions:

1) Mathematical operations performed once at the beginning of the program which are described below:

   a) Conversion of input data from engineering units to ft-lb-sec. units.

   b) Assembly of the transpose of the connection matrix A^T from CONP(J) and CONN(J) arrays defined later in the description of input data.

   c) Transposition of the transpose of the connection matrix A^T, to obtain the connection matrix A.

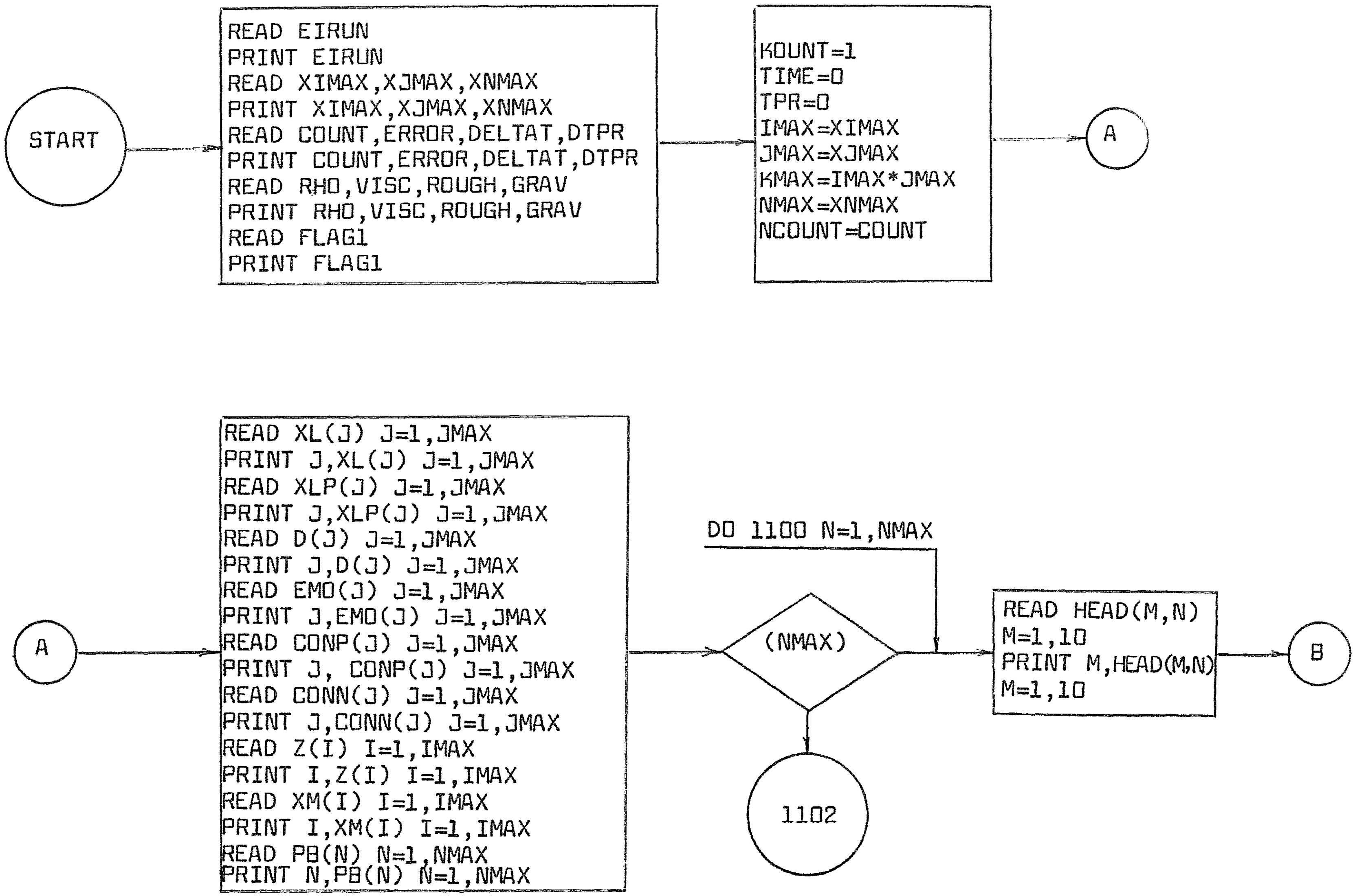


Figure 4
Flow Chart of HYTRAN MAIN Program

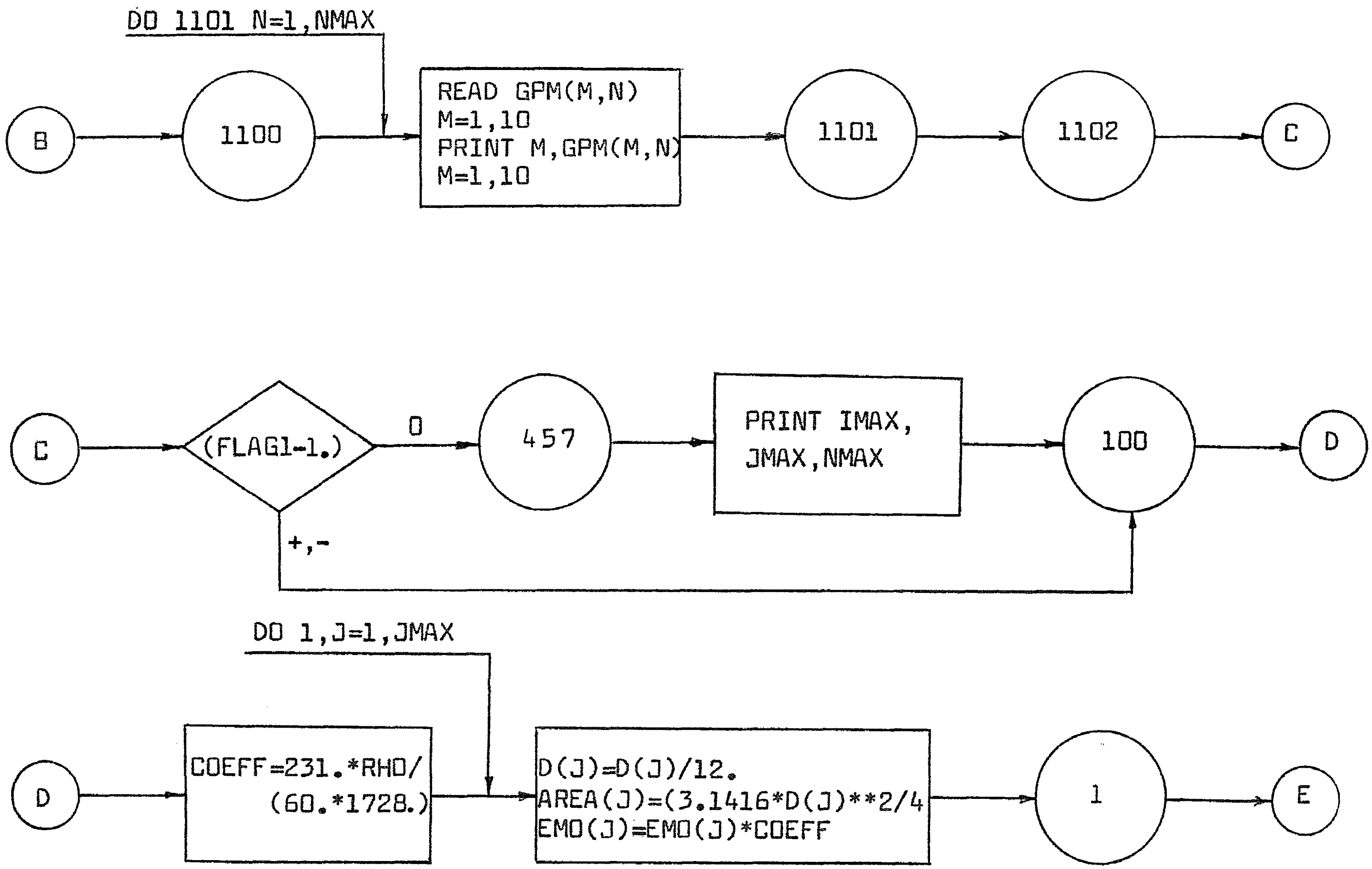


Figure 5
Flow Chart of HYTRAN MAIN Program

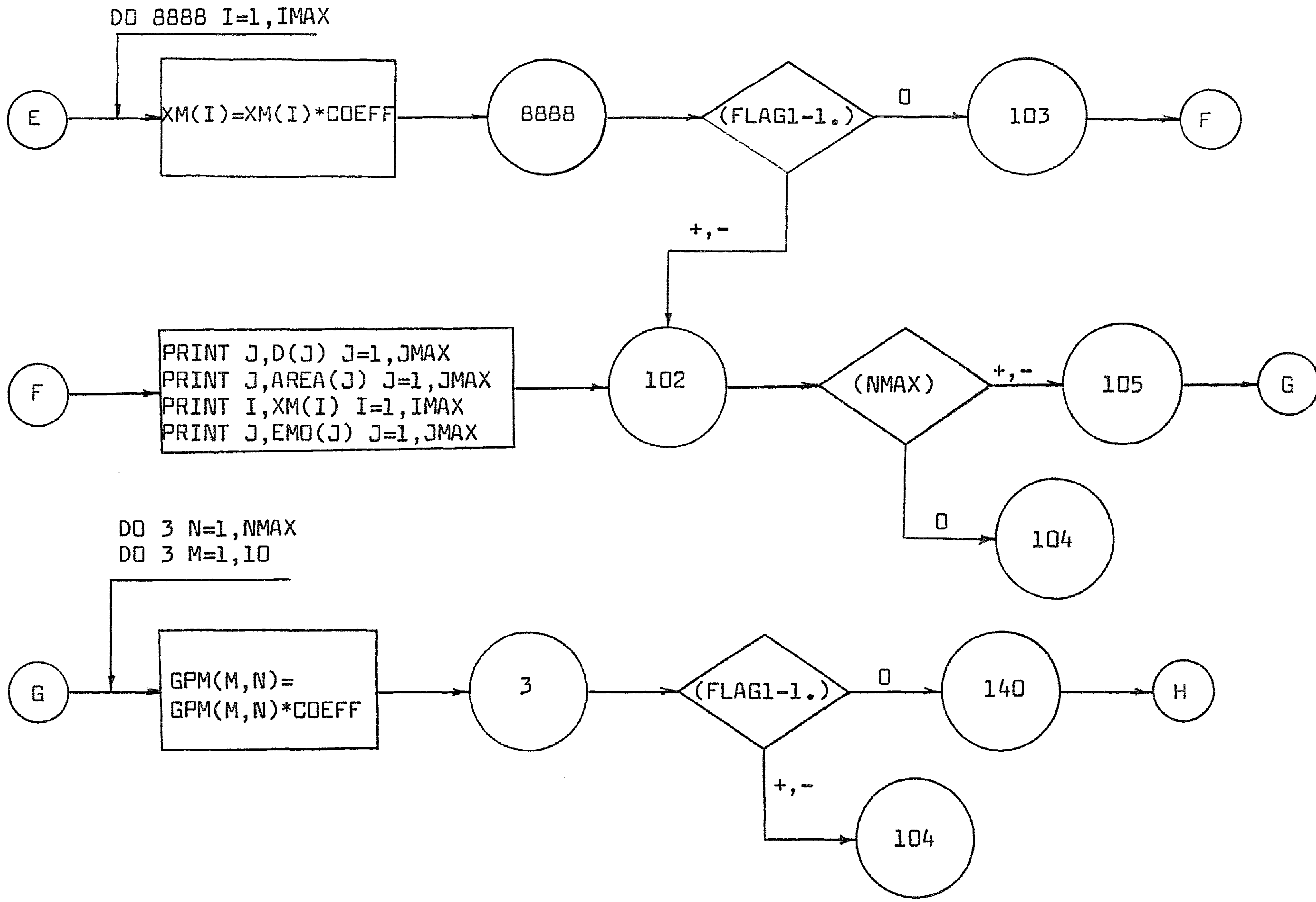


Figure 6
Flow Chart of HYTRAN MAIN Program

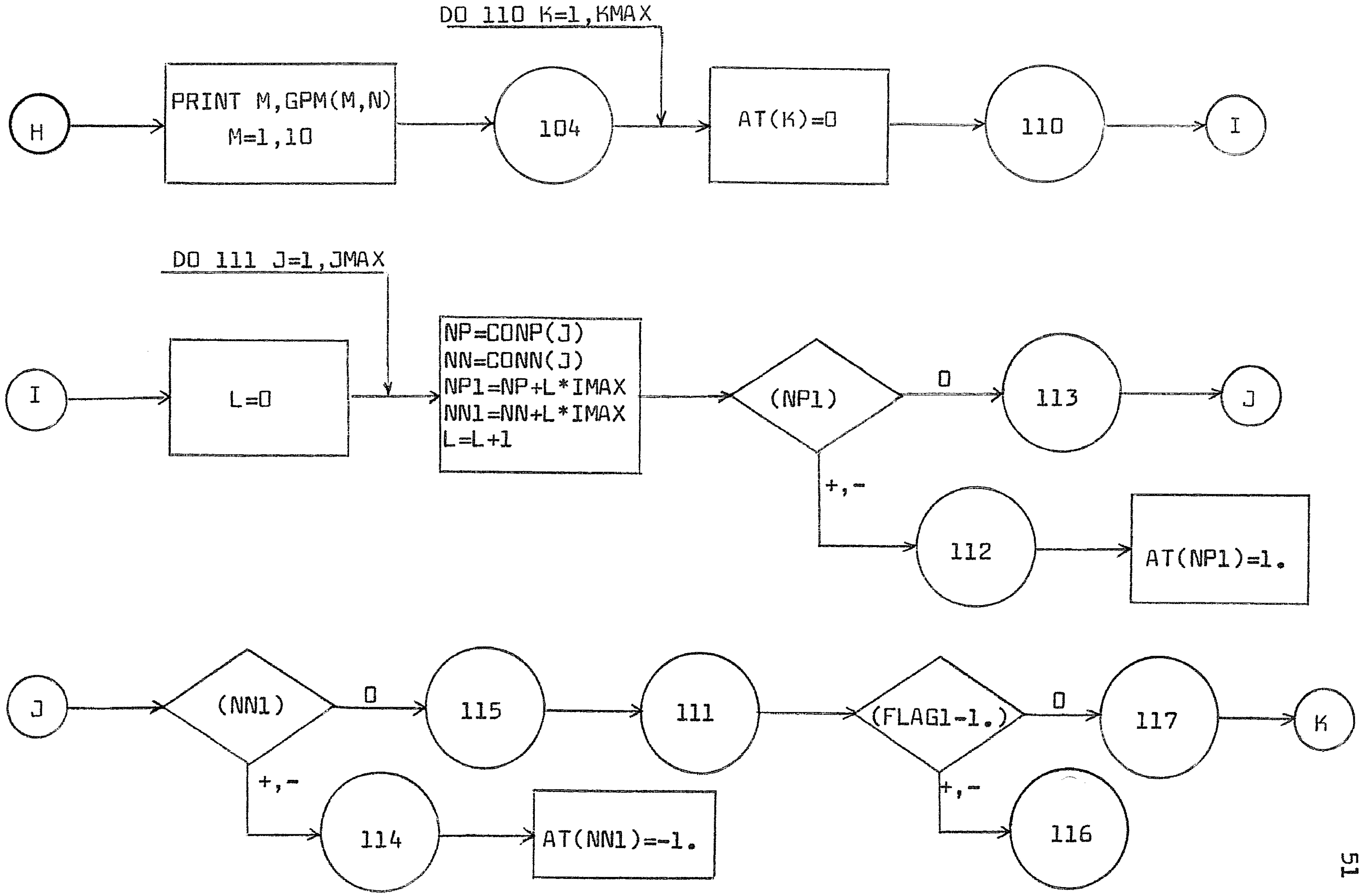


Figure 7
Flow Chart of HYTRAN MAIN Program

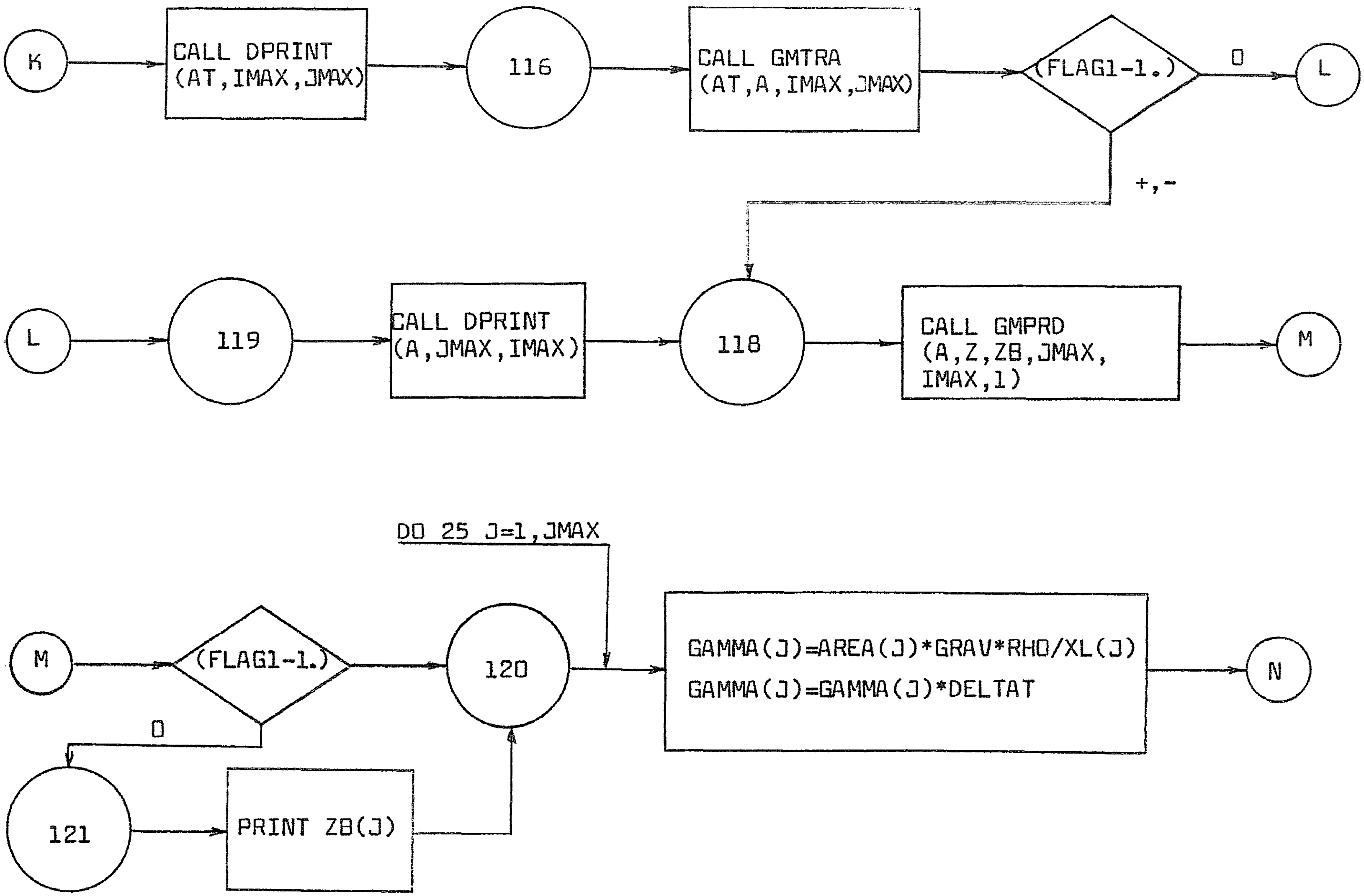


Figure 8
Flow Chart of HYTRAN MAIN Program

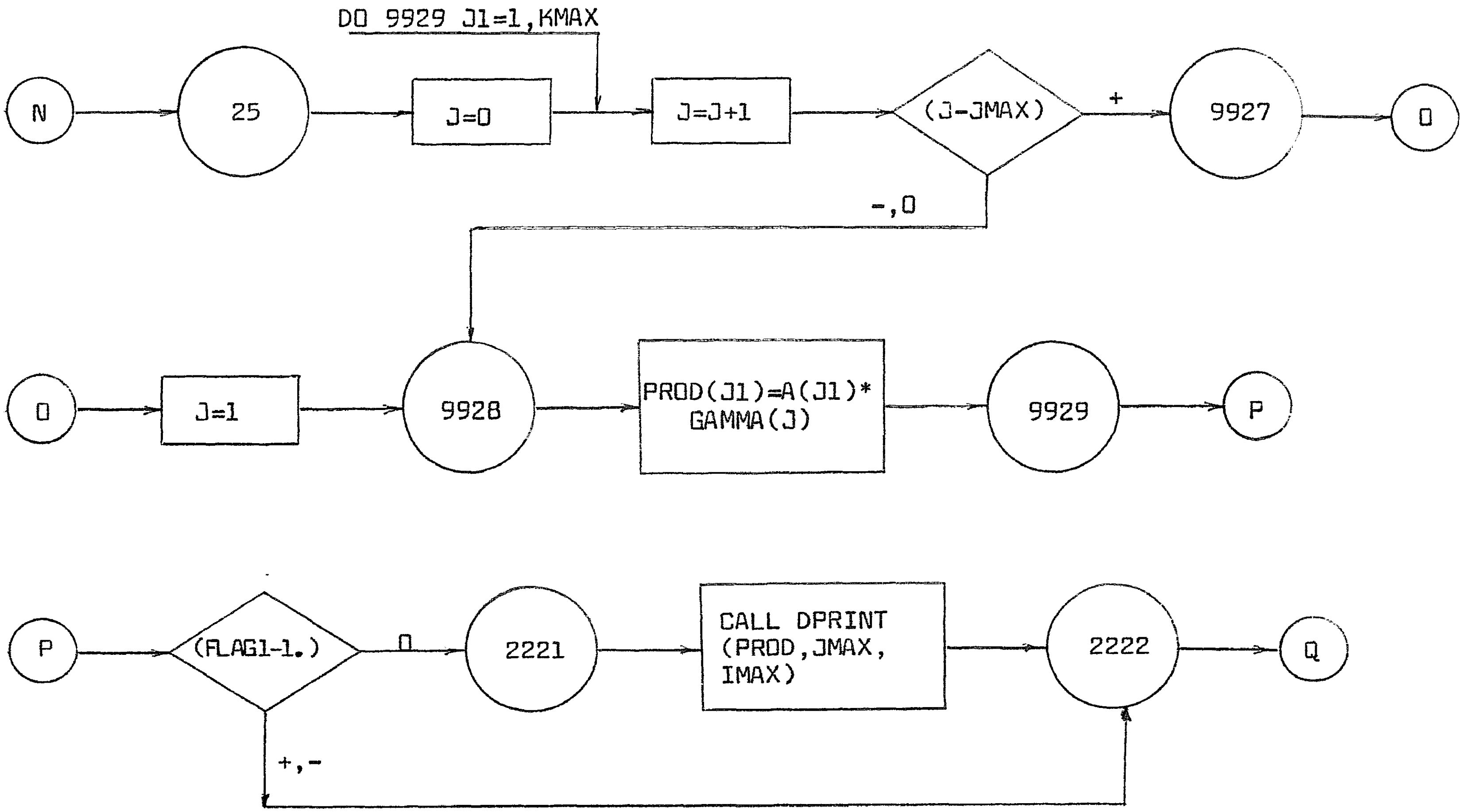


Figure 9
Flow Chart of HYTRAN MAIN Program

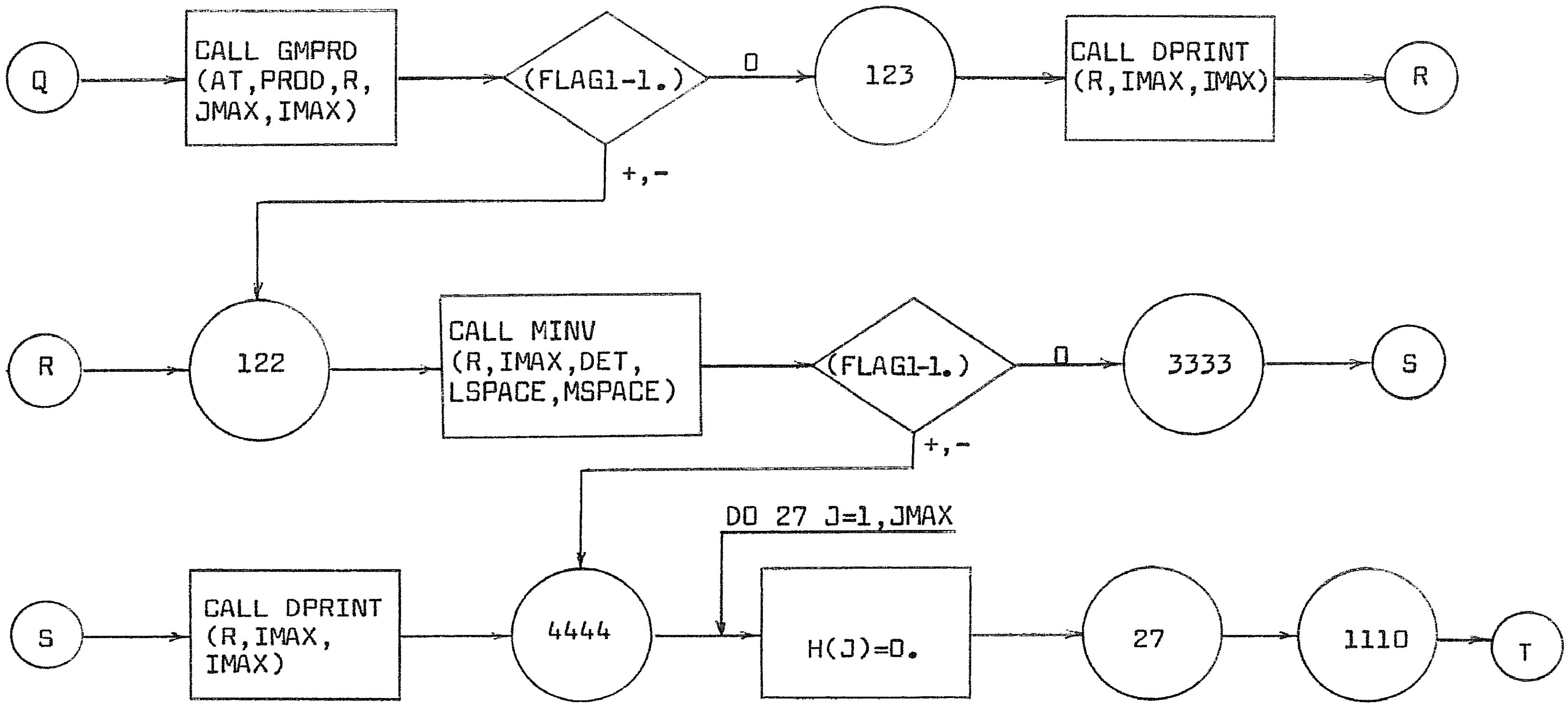


Figure 10

Flow Chart of HYTRAN MAIN Program

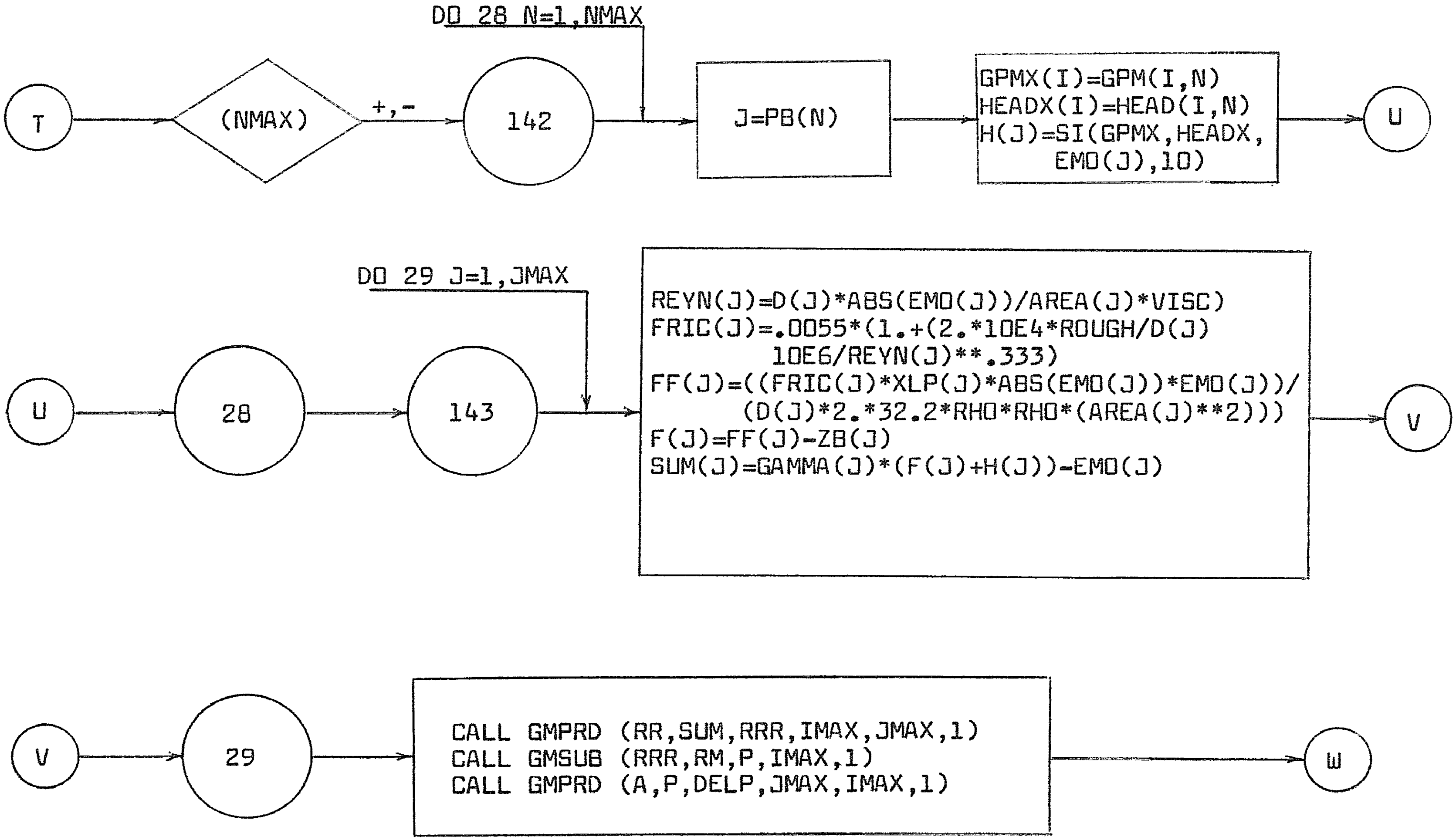


Figure 11

Flow Chart of HYTRAN MAIN Program

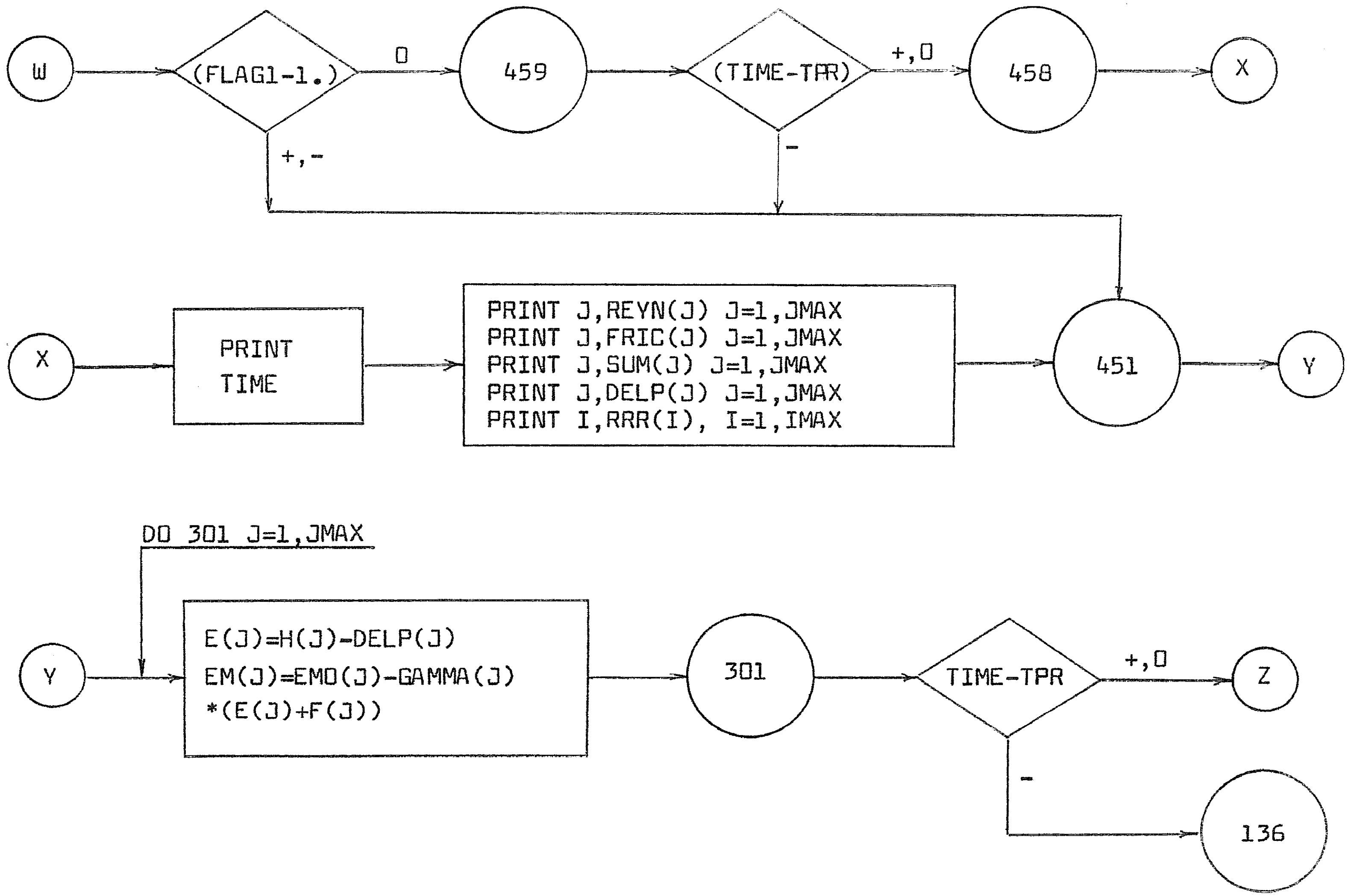


Figure 12

Flow Chart of HYTRAN MAIN Program

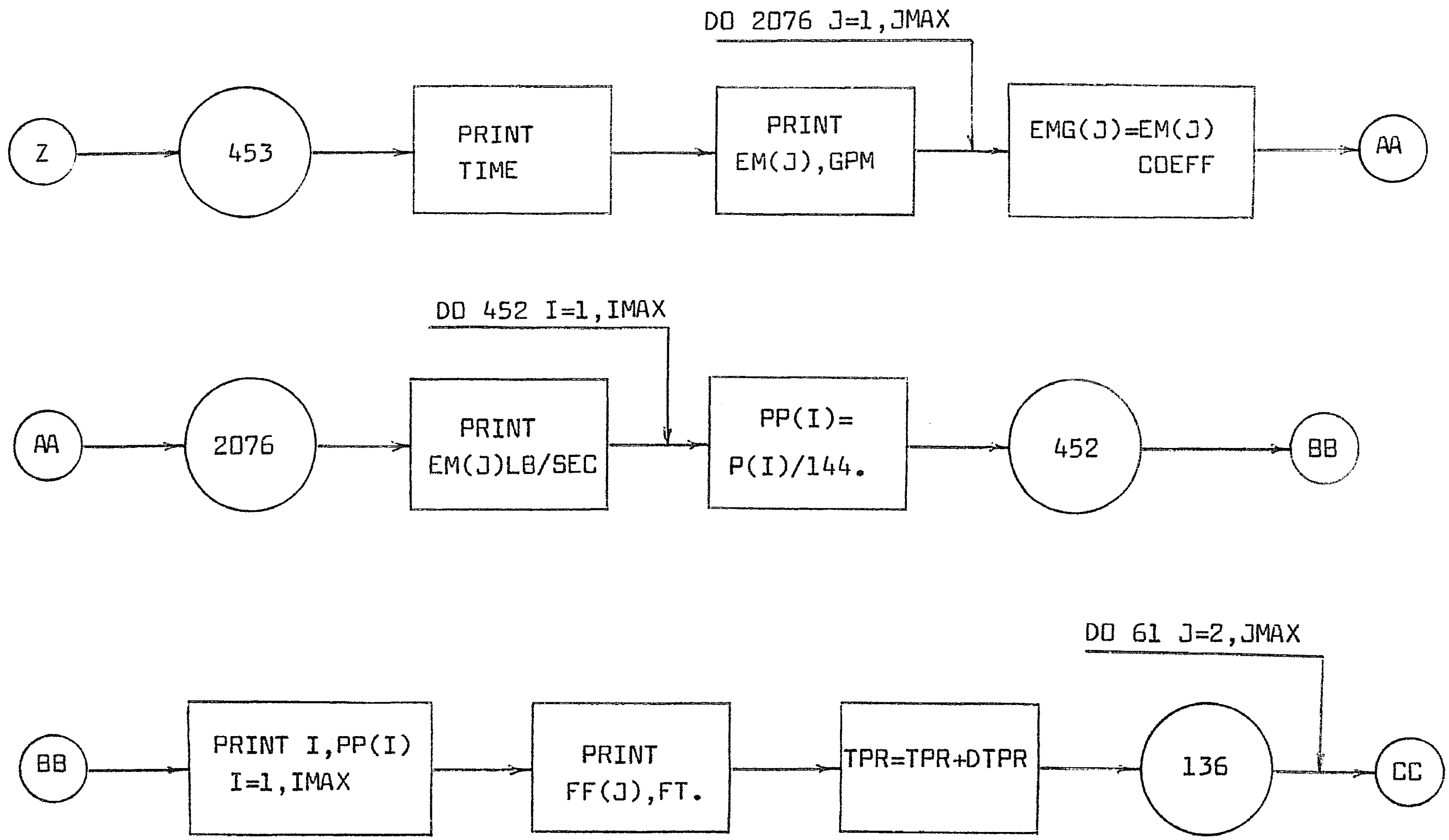


Figure 13

Flow Chart of HYTRAN MAIN Program

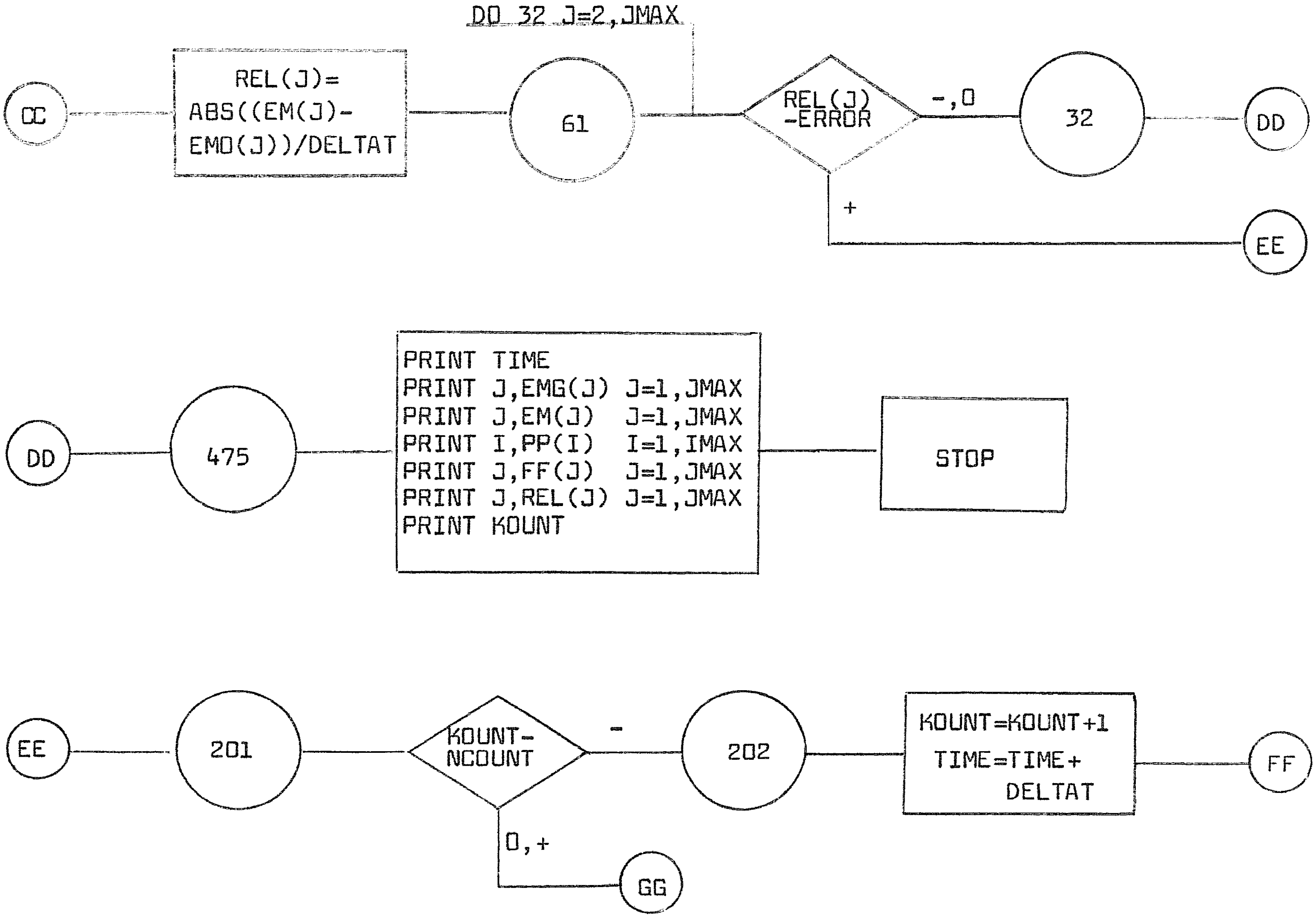


Figure 14

Flow Chart of HYTRAN MAIN Program

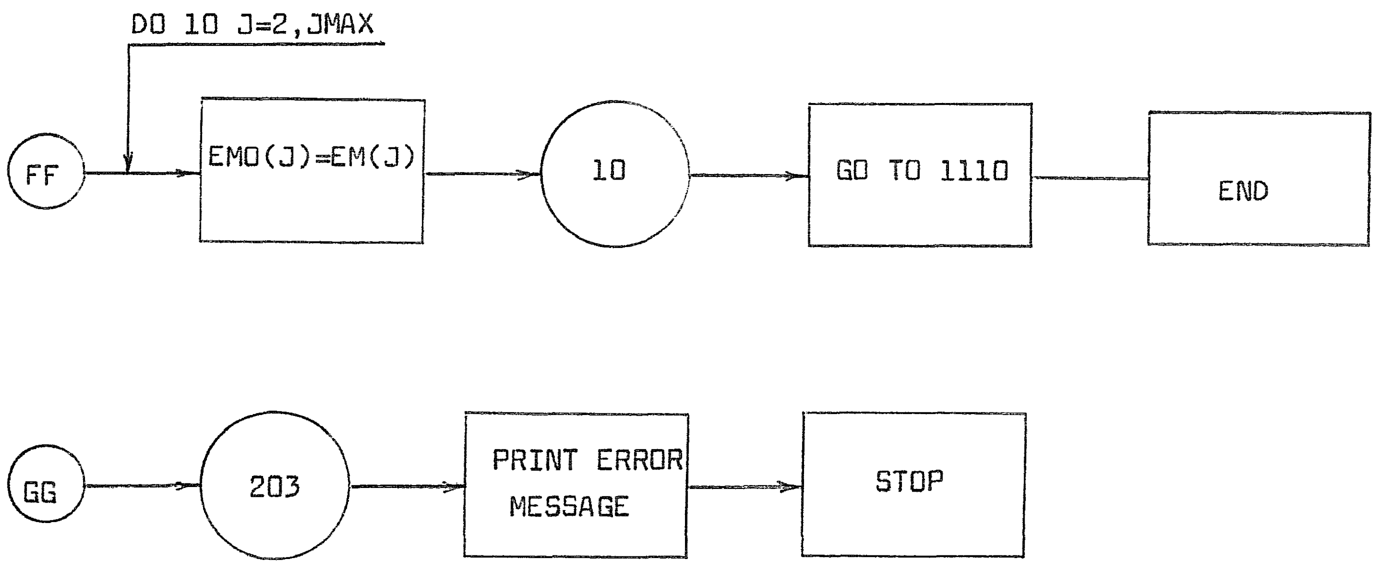


Figure 15

Flow Chart of HYTRAN MAIN Program

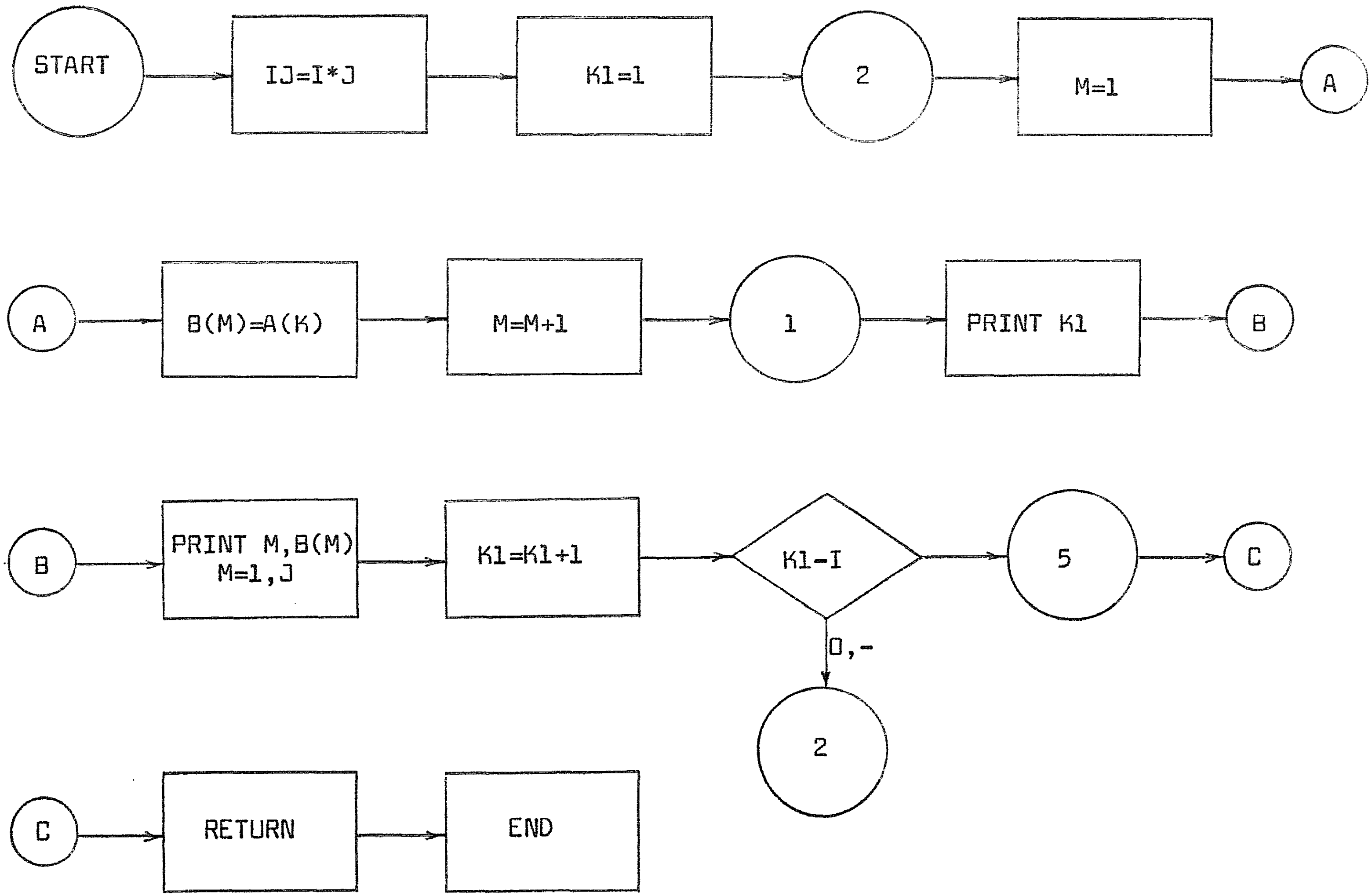


Figure 16
Flow Chart of Subroutine DPRINT

d) Multiplication of the connection matrix A by nodal elevation array Z to obtain branch height difference ΔZ.

e) Formation of matrix M defined by equation (9).

f) Formation of products $M^{-1} \cdot A^T$ and $M^{-1} \cdot \dot{m}_{io}$ required by equation (8).

2) Iterative operations performed after each time increment as described below:

a) Calculation of branch static, transducer and frictional heads and their multiplication by γ.

b) Summation of the above terms and the present values of the branch mass flow rates required by equation (8).

c) Multiplication of $M^{-1} \cdot A^T$ by the result obtained under(b) above and subtraction of $M^{-1} \cdot \dot{m}_{io}$ from this product to form the pressure array p as indicated by equation (8).

d) Calculation of branch pressure drop array Δp from equation (1).

e) Calculation of branch mass flow rate array from equation (6).

MINV Matrix Inversion Subroutine

This subroutine is used to invert matrices. It is used only once at the beginning of the program to invert matrix M designated by equation (9). MINV is a standard subroutine from reference (22) and its calling sequence is:

CALL MINV (A,N,D,L,M)

A = Input matrix, destroyed in computation and replaced by resultant inverse.

N = Order of matrix A

D = Resultant determinant (not used in this analysis)

L = Work vector of length N

M = Work vector of length N

GMPRD Matrix Product Subroutine

This subroutine is used for the multiplication of matrices, as discussed above. It is a standard subroutine from reference (22) and its calling sequence is:

CALL GMPRD (A,B,R,N,M,L)

A = Name of first input matrix

B = Name of second input matrix

R = Name of output matrix

N = Number of rows in A

M = Number of columns in A and rows in B

L = Number of columns in B

GMSUB Matrix Subtraction Subroutine

This subroutine is used for the subtraction of two matrices. It occurs once in the program to perform the matrix subtraction indicated by equation (8). It is a standard subroutine from reference (22) and its calling sequence is:

CALL GMSUB (A,B,R,N,M)

A = Name of first input matrix

B = Name of second input matrix

R = Name of output matrix

N = Number of rows in A, B and R

M = Number of columns in A, B and R

<u>GMTRA Matrix Transposition Subroutine</u>

This subroutine is used to transpose matrices. It is used once in the program to transpose the connection matrix A to A^T. It is a standard subroutine from reference (<u>22</u>) and its calling sequence is:

CALL GMTRA (A,R,N,M)

A= Name of matrix to be transposed

R = Name of resultant matrix

N = Number of rows of A and columns of R

M = Number of columns of A and rows of R

<u>SI FUNCTION Linear Interpolation Subroutine</u>

This function is used to interpolate values in the head versus volumetric flow tables for transducers (pumps). It is a standard subroutine and its calling sequence is:

FUNCTION SI (XTBL, YTBL, X, N)

XTBL = Independent variable table

YTBL = Dependent variable table

X = Value of independent variable

N = Number of points in table

DPRINT Matrix Printout Subroutine

This subroutine, charted on Figure 16, is used for printing the following matrices: A^T, A, M and its inverse. These matrices can be printed at the users option, for debugging purposes, by setting appropriate values for FLAG 1 as shown in the description of input data. The calling sequence of this subroutine is:

```
CALL DPRINT (A,I,J)
```

A = Name of matrix to be printed

I = Number of rows in matrix

J = Number of columns in matrix

II. DESCRIPTION OF INPUT DATA

The input data can be divided into two groups:

1) Non-subscripted variables

2) Subscripted variables

A brief description of each input variable appears in the program listing to aid in checking the program. A more detailed description of input data is presented in this section. The procedure used for labeling the input data is described below:

All non-subscripted variables are identified by the prefix "FIX", followed by two numbers. The first number indicates the card number and the second indicates the order of the variable on that card. For example, FIX 2-3 XNMAX denotes that the variable XNMAX is a non-subscripted variable appearing as the third value on card number 2. All subscripted variables are identified by a prefix "VAR" followed by one number. For example, VAR 7 Z denotes that Z is a subscripted variable appearing as the seventh set of arrays in the input data. Each card contains 6 values of Z(I). All input data are entered using the FORMAT E10.3.

Non-subscripted Variables

FIX 1-1 EIRUN Run Number. This quantity is used for identification of successive computer runs.

FIX 2-1 XIMAX Maximum Number of Nodes. This number represents the maximum number of nodes in the system being analyzed. These nodes are the points on Figure 1 denoted by the non-encircled numbers.

FIX 2-2 XJMAX Maximum Number of Branches. This value represents the maximum number of branches in the system being analyzed. These are the lines on Figure 1 denoted by the encircled numbers. As can be noted from Figure 1, the branch numbers do not have to correspond with node numbers.

FIX 2-3 XNMAX Maximum Number of Transducers (Pumps). This quantity describes the maximum number of transducers (pumps) in the system being analyzed. It should be noted that any flow device or restrictions whose flow versus head characteristics are known can be included in this category.

FIX 3-1 COUNT Maximum Allowable Number of Iterations. This number represents the maximum number of iterations allowed in the analysis and is a safety device to stop the computer in the event of program error. The value of COUNT is calculated by

$$\text{COUNT} = \frac{\text{Maximum problem physical time}}{\text{DELTAT}}$$

Since the maximum problem physical time is not known before the problem solution, a suggested value of COUNT is 1000. If more transient time is required, an error message will appear on the output sheet and COUNT should then be increased. Steady-state solutions generally use large time increments and require fewer iterations. Thus, the value of COUNT = 1000 is generally adequate for steady-state analyses.

FIX 3-2 ERROR Steady-State Flow Convergence Error. This value represents the steady-state flow convergence error. Steady-state conditions are considered to be reached when

$$\frac{\dot{m} - \dot{m}^{o}}{h} \leq \text{ERROR}$$

FIX 3-3 DELTAT Time Increment, Sec. This variable specifies the time increment for each integration step. It is equivalent to h, presented earlier in the text. The criterion for its selection is discussed in the section on "Numerical Stability Analysis". The time increment is selected by first calculating all branch time constants using equation (14) and then setting the time increment equal to 1/20 of the minimum time constant as specified by equation (15).

FIX 3-4 DTPR Printout Interval, Sec. This number specifies the desired time interval for the printout of the transient solution. It is used to avoid the printing of results after every iteration and minimize computer running time. For the transient case, a suggested value of 10 times DELTAT proved satisfactory for this quantity. For the steady-state solution when only the final converged value is desired, DTPR should be of the order 1000 times DELTAT.

FIX 4-1 RHO Fluid Density lbs/cu.ft. This number specifies the density of the fluid flowing in the hydraulic network.

FIX 4-2 VISC Fluid Viscosity lbs/ft-sec. This number represents the viscosity of the fluid in the system.

FIX 4-3 ROUGH Pipe Absolute Roughness, ft. This quantity represents the absolute roughness of pipes used in the network. The program divides this quantity by the hydraulic diameter to obtain relative roughness.

FIX 4-4 GRAV Gravitational Acceleration ft/sec^2. This number represents the force of gravity acting on the system under investigation (32.2 ft/sec^2).

FIX 5-1 FLAG 1 Printout Indicator. This parameter is used primarily for debugging purposes. It controls the printing of the following variables defined later in the description of the output data.

IMAX, JMAX, NMAX, D(J), AREA(J), XM(I), EMO(J), GPM(M,N), A^T, A, ZB(J), PROD(J1), R, REYN(J), FRIC(J), F(J), SUM(J), RR, E(J), RRR, P(I), DELP(J)

Assigning a value of 1 to FLAG 1 prints the above variables and a value of 0 eliminates printing.

Subscripted Variables

VAR 1 XL(J) Branch Actual Length, Ft. These values represent the actual developed length of each pipe branch.

VAR 2 XLP(J) Branch Equivalent Length, Ft. These values define the branch equivalent length composed of actual length XL(J) plus the resistance due to valves, fittings, expansions and contractions and other flow restrictions expressed in terms of pipe length.

VAR 3 D(J) Branch Hydraulic Diameter, Inches. These values represent the hydraulic diameter of each branch. The hydraulic diameter for round pipes is equal to their actual inside diameter. The hydraulic diameter for pipes other than circular cross section is equal to four times their volume divided by their wetted surface.

VAR 4 EMO(J) Branch Initial Mass Flow Rate, GPM. These values define the initial mass flow rates in each branch. For steady-state analysis, these values are assumed without any restriction except that continuity equations must be satisfied at each node. For transient analysis, the prevailing steady-state values should be entered. These values may be obtained either from a previous run on this program or from other sources.

VAR 5 CONP(J) Positive Branch Node, VAR 6 CONN(J) Negative Branch Node. A node is defined as any point in the system at which three or more flows meet or network geometric dimensions change. A branch is defined as a line connecting two nodes. The nodes and the branches of the hydraulic network are numbered separately starting from 0 for nodes and 1 for branches without skipping any number. The node numbered 0 should correspond to the point in the system where the pressure is known. The branch number 1 corresponds to the branch connected to node 0. Every branch in the system is given an arbitrary positive orientation or direction designated by an arrow. The flow is considered positive when it is along this positive direction and negative when it is against this positive direction. The node located at the tail of the arrow is designated as the "sending node" for that branch and the node located at the head of the arrow is designated as the "receiving node". The value of CONP(J) and CONN(J) represent the node number corresponding to branch J for the sending node and the receiving node, respectively. For example, the values of CONP(J) and CONN(J) for J =36 and J = 37 in Figure 1 will become:

	J = 36	J = 37
CONP(J)	9	8
CONN(J)	10	9

The CONP(J) and CONN(J) arrays are used by the program to assemble the connection matrix. All CONP(J) nodes contribute a value of +1 to the connection matrix, and all CONN(J) nodes contribute a -1 to the connection matrix.

VAR 7 Z(I) Node Elevation, Ft. These values denote the algebraic value of height of each node in the system with respect to node 0 taken as a datum.

VAR 8 XM(I) Nodal Mass Flow Rate, GPM. These values represent flow rates entering or leaving the system nodes. Flow entering a system node is designated as negative. Flow leaving the system node is designated as positive. In an open system, these flows represent the network load and are known. In a closed system, these values are zero and must be entered as such.

VAR 9 PB(N) Branch Locations of Transducers (Pumps). These variables define the branch location of the transducers in the system. A typical input data card would appear as follows:

7 10 12 22 37 4

This would indicate that transducer number 1 is located in branch 7, transducer number 2 is located in branch 10,...etc. If there are no pumps in the system, this array may be omitted by setting XNMAX equal to zero. This will omit the entry of these values.

VAR 10 HEAD(M,N) Transducer Head, Ft. This array specifies the head variation for each transducer in the system. The value of M represents the running index for the points in the table for transducer number N.

VAR II GPM(M,N) Transducer Volumetric Flow Rate, GPM. This array specifies the volumetric flow rate variation for each transducer in the system. These values together with those in VAR 10 define the flow versus head characteristics of the transducer. The value of transducer head used in the analysis is determined by the interpolation subroutine (SI FUNCTION) described earlier.

III. DESCRIPTION OF OUTPUT DATA

The output data can be conveniently divided into the following groups:

1) Input Data Printout

All input data are printed out and labeled using variable names defined in the description of input data.

2) Debugging Printout

At the users option, under the control of FLAG 1, described in the input data, the following calculated variables may be printed out for debugging or checking purposes. These data are classified into two groups:

a) Debugging printout for mathematical operations performed once at the beginning of the program labeled as indicated below:

IMAX, JMAX, NMAX

These values are equivalent to XIMAX, XJMAX and XNMAX, described in the input data, after being converted to fixed mode.

D(J) Branch Hydraulic Diameter, Ft.

These values represent the branch hydraulic diameters after being converted to feet.

AREA(J) Branch Cross-Sectional Area, Ft.^{2}

These values specify the calculated branch flow cross-sectional area.

XM(I), EMO(J), GPM(M,N)

These quantities represent mass flow rates, as defined previously in the description of input data, after conversion from GPM to lb/sec.

A^T Transpose of System Connection Matrix

This matrix is generated using the input variables CONN and CONP and is printed under subroutine DPRINT.

A System Connection Matrix

This matrix is obtained by transposing matrix A^T, using GMTRA subroutine.

ZB(J) Branch Elevational Difference Array, Ft.

These values represent the elevational difference measured along the positive flow direction. For example, referring to Figure 1, the branch elevational difference for branch 38 is Z(18)-Z(12).

PROD Matrix

This matrix is the product of the following two matrices--the first matrix having branch γ's along its diagonal and zero elsewhere, and the other, the system connection matrix A, as indicated by equation (9).

R Matrix

This matrix is the product of the transpose of the system connection matrix A^T times the PROD matrix, represented by matrix M in equation (9).

R^{-1} Matrix

This matrix is the inverse of the R matrix defined above.

RR Matrix

This matrix is defined as the product of the inverse of the R matrix times the transpose of the connection matrix A^T as required by equation (8).

RM Array

These values constitute an array obtained by multiplying the inverse of the R matrix times the XM array defined earlier.

b) Debugging printout for mathematical operations performed after each time integration labeled as indicated below:

REYN(J) Reynolds Number

Reynolds Numbers are calculated as the product of hydraulic diameter times mass flow rate divided by the product of the flow cross-sectional area times fluid viscosity. They are used to determine friction factors in the system branches.

FRIC(J) Friction Factor

These quantities are calculated from equation (5) based on Moody's Method to determine frictional head loss in pipes.

SUM(J) Array

These values denote an intermediate calculation and represent the quantity contained in the first set of braces on the right-hand side of equation (8).

DELP(J) Total Branch Pressure Difference lb/ft^2

These values represent the total branch pressure difference computed from equation (1).

RRR Array

These values constitute an array obtained by multiplying RR matrix by SUM array, both defined earlier.

3) Variables Defining the Problem Solution

These variables are printed after every DTPR seconds and define the problem transient solution. They are labeled as indicated below:

EM(J)	Branch Mass Flow Rate, GPM
EM(J)	Branch Mass Flow Rate, lb/sec
P(I)	Nodal Pressures, lb/in^2
FF(J)	Branch Frictional Pressure Drop, Ft

4) Final Steady-State Printout

The variables are printed only when steady-state conditions are reached. They are labeled as indicated below.

REL(J) Steady-State Convergence Error

These values represent the slope of the mass flow rate time response curve at steady-state conditions.

KOUNT Actual Number of Iterations

This value represents the actual number of iterations performed in order to reach steady-state conditions.

In addition to REL(J) and KOUNT, all information stated under paragraph (3) above is also printed out.

IV. OPERATING PROCEDURE

To employ the HYTRAN computer program effectively, the user should familiarize himself with sections on the mathematical formulation, numerical stability analysis, and the presentation of results. He should then prepare a diagram of the system to be analyzed and assign a separate numbering system to nodes and branches as shown in Figure 1. The nodal point in the system where the pressure is known is numbered as node zero and the remaining nodes are numbered arbitrarily without skipping any number. It should be emphasized that node zero in the hydraulic network simulates a reservoir with known pressure and infinite water capacity. The algebraic sum of all XM(I) will enter the system at node zero and the nodal pressures in the system are determined with respect to the reservoir pressure considered as datum.

The user should then prepare the input data in accordance with the format and description given in part two, section II. These input data are then punched on cards and assembled with the HYTRAN program together with a number of control cards. These control cards vary from one computer organization to the next. A sample of the deck assembly for RCA Spectra-70 is given below:

```
                          run time sec, no of output lines
// JOB........., HYTRAN,.................................,
// PARAM LIST = YES, DEBUG = YES
//  FORTRN
HYTRAN PROGRAM DECK
// EXEC
INPUT DATA DECK
```

After the program has been run, the user should verify the following points:

1) Correctness of Input Data and KOUNT Error Message

The input data printed on output sheets should be verified for possible errors in punching. Furthermore, the output sheets should be examined for a possible error message on KOUNT and proper action taken to rectify this error condition as described in the description of input data.

2) Convergence of the Transient Solution

The branch mass flow rates are then examined and plotted. If the integration time step is properly selected according to the numerical stability analysis described earlier, the time variation of the branch flow rate will be smooth and will not exhibit an oscillating pattern. However, since the numerical stability criterion depends on the approximate values of steady-state mass flow rates, and these values are not exactly known before the problem solution, the mass flow rate in the branch with the smallest time constant may at times become oscillatory or non-convergent. When such a situation occurs, one should, using the run performed, adjust the estimate of steady-state mass flow rates to decrease the branch time constants and therefore reduce the integration time step. To verify the convergence of the transient solution, the user may employ one of the following techniques:

a) Determine the integration time increment based on steady-state mass flow rates. If this integration time increment is close to the one initially employed, the solution has converged.

b) Enter a value of time increment equal to one half of the previous time increment and compare the transient results on mass flow rates.

When the user is not interested in the transient solution and he wishes to obtain only the final steady-state solution, the time increment can be increased to the order of the system minimum time constant or even higher, leading to a large reduction in computer running time.

3) Convergence of the Final Steady-State Solution

The convergence of the final steady-state values can be determined by examining the branch mass flow rate curves plotted earlier. If these curves approach asymptotically to constant steady-state values, the final steady-state solution has been reached. If not, the value of the steady-state convergence error (ERROR) should be reduced.

V. HYTRAN PROGRAM NOMENCLATURE

To assist the user in the modification of the HYTRAN program, a list of some key variables with their corresponding analysis notation and their definition is presented in this section.

Program Notation	Analysis Notation	Description
A	A	Connection matrix for branches and nodes as defined in the text
AREA	a	Branch flow cross-sectional area, ft^2
D	D	Branch hydraulic diameter, ft
FF	F	Pressure loss due to friction, lb/ft^2
FRIC	f	Friction factor
GRAV	g	Acceleration of gravity, 32.2 ft/sec^2
H	H	Transducer pressure differential (pump pressure rise), ft
DELTAT	h	Time increment, sec
XL	L	Branch actual length, ft
XLP	L	Branch equivalent length (Branch actual length plus additional lengths allowed for fittings), ft
R	M	A matrix defined by equation (9)
EM	$\dot{m}$	Updated value of branch mass flow rate, lb/sec
EMO	$\dot{m}^o$	Present value of branch mass flow rate, lb
XM	$\dot{m}_{io}$	Nodal source and sink (input and output) mass flow rate (positive for source and negative for sink), lb/sec
P	p	Nodal pressure, psi
REYN	R_n	Branch Reynolds Number ($= D\,\dot{m}^o/a\mu$)

Program Notation	Analysis Notation	Description
TIME	t	Time, sec
Z	Z	Nodal elevations, ft
GAMMA	γ	A quantity defined in the text
DELP	Δp	Pressure drop along the positive flow direction, psi
ZB	ΔZ	Branch elevational difference along the positive flow direction (elevation of node at the tail of the arrow minus elevation of the node at the head of the arrow), ft
ROUGH	ϵ	Pipe roughness, ft
VISC	μ	Fluid viscosity, lb/sec - ft
RHO	ρ	Fluid density, lb/ft^3
ERROR	-	Steady-state convergence error

VI. HYTRAN PROGRAM LISTING

The HYTRAN program is written in FORTRAN-IV for RCA Spectra Model 70/45 digital computer. A listing of the HYTRAN program appears in the following pages.

FORTRAN IV030 SOURCE PROGRAM

```
      PROGRAM HYTRAN
C           MAIN PROGRAM  WATER NETWORK FIXED GEOMETRY TRANSIENT ANALYSIS
C                                           G.V.CATANZARO GRAD ME
      DIMENSION A(2000),XL(50),XLP(50),Z(40),D(50),PB(5),HEAD(10,5),
     1              GPM(10,5),EMO(50),XM(40),GPMX(10),HEADX(10)
      DIMENSION AT(2000),ZB(50),GAMMA(50),PROD(2000),R(1600),LSPACE(40),
     1            MSPACE(40),H(50),REYN(50),FRIC(50),F(50),SUM(50),RR(2000),
     2              RRR(40),P(40),DELP(50),EM(50),CONP(50),CONN(50),FEL(50),
     3              AREA(50),E(50),FF(50),EMG(50),PP(40),RM(40)
C
C  FIX NO.   NOTATION         DESCRIPTION                         UNITS
C
C  1-1       EIRUN            RUN NO.DESIGNATION
C
C  2-1       XIMAX            MAX.NO.OF NODES
C  2-2       XJMAX            MAX.NO.OF BRANCHES
C  2-3       XNMAX            MAX.NO.OF PUMPS
C
C  3-1       COUNT            MAX.ALLOW.NO OF ITERATIONS
C  3-2       ERROR            STEADY STATE CONVERGENCE ERROR
C  3-3       DELTAT           TIME INCREMENT                      SEC.
C  3-4       DTPR             PRINTOUT INTERVAL
C
C  4-1       RHO              FLUID DENSITY                       LBM/CU-FT
C  4-2       VISC             FLUID VISCOSITY                     LBM/FT*SEC
C  4-3       ROUGH            PIPE ABSOLUTE ROUGHNESS             INCHES
C  4-4       GRAV             GRAVITATIONAL ACCELERATION          FT/SEC**2
C
C  5-1       FLAG1            PRINTOUT  INDICATOR-1 PRINTS IMAX,JMAX,NMAX,
C                             D(J),AREA(J),XM(I),EMO(J),GPM(M,N),AT,A,ZB,
C                             PROD(J1),R,REYN(J),FRIC(J),F(J),SUM(J),RR,E(J)
C                             RRR,P,  0 SKIPS PRINT
C
C  VAR.NO.   NOTATION         DESCRIPTION                             NO.OF CARDS
C  1         XL(J)            BRANCH ACTUAL LENGTH,FT.                XJMAX/6
C  2         XLP(J)           BRANCH EQUIVALENT LENGTH,FT.            XJMAX/6
C  3         D(J)             BRANCH DIAMETER,INCHES                  XJMAX/6
C  4         EMO(J)           BRANCH INITIAL MASS FLOW RATE,GPM       XJMAX/6
C  5         CONP(J)          POSITIVE BRANCH NODE                    XJMAX/6
C  6         CONN(J)          NEGATIVE BRANCH NODE                    XJMAX/6
C
C  7         Z(I)             NODE ELEVATION,FT                       XIMAX/6
C  8         XM(I)            NODE FLOW,GPM                           XIMAX/6
C
C  9         PB(N)            BRANCH NO. OF PUMPS                     XNMAX/6
C  10        HEAD(M,N)        PUMP HEAD,FT                            2*XNMAX
C  11        GPM(M,N)         PUMP VOLUMETRIC FLOW,GPM                2*XNMAX
C***************************************************************************
      NIN=5
      NOUT=6
```

```
9999 FORMAT(10X,I4)
1000 FORMAT(2X,6(6X,E14.7))
1001 FORMAT(1E10.3)
1002 FORMAT(2E10.3)
1003 FORMAT(3E10.3)
1004 FORMAT(4E10.3)
1005 FORMAT(5E10.3)
1006 FORMAT(6E10.3)
1007 FORMAT(6(I5,1X,E14.7,2X))
C       ***READ AND PRINT NONSUBSCRIPTED VARIABLES***
        WRITE(NOUT,2054)
2054 FORMAT('1',14X,('INPUT DATA                                      HYTRAN PROGRAM')
       1,/)
        READ(NIN,1001)EIRUN
        WRITE(NOUT,2001)
2001 FORMAT(1H012X,'EIRUN')
        WRITE(NOUT,1000) EIRUN
C
        READ(NIN,1003)XIMAX,XJMAX,XNMAX
        WRITE(NOUT,2002)
2002 FORMAT(1H0,14X,'XIMAX                 XJMAX                          XNMAX')
        WRITE(NOUT,1000)XIMAX,XJMAX,XNMAX
C
        READ(NIN,1004) COUNT,ERROR,DELTAT,DTPR
        WRITE(NOUT,2003)
2003 FORMAT(1H0,14X,'COUNT                 ERROR                      DELTAT
       1          DTPR')
        WRITE(NOUT,1000)COUNT,ERROR,DELTAT,DTPR
C
        READ(NIN,1004)RHO,VISC,ROUGH,GRAV
        WRITE(NOUT,2004)
2004 FORMAT(1H0,14X,'RHO                   VISC                         ROUGH
       1            GRAV')
        WRITE(NOUT,1000)RHO,VISC,ROUGH,GRAV
C
        READ(NIN,1001)FLAG1
        WRITE(NOUT,2005)
2005 FORMAT(1H0,14X,'FLAG1')
        WRITE(NOUT,1000)FLAG1
        KOUNT=1
        TIME=0.
        TPR=0.
C
C
C
        IMAX=XIMAX
        JMAX=XJMAX
        KMAX=IMAX*JMAX
        NMAX=XNMAX
        NCOUNT=COUNT
```

FORTRAN IV030 SOURCE PROGRAM HYTRAN PROGRAM

```
C         ***READ AND PRINT SUBSCRIPTED VARIABLES***
          READ(NIN,1006) (XL(J),J=1,JMAX)
          WRITE(NOUT,2006)
     2006 FORMAT(1H0,10X,'XL(J)',/)
          WRITE(NOUT,1007) (J,XL(J),J=1,JMAX)
C
          READ(NIN,1006) (XLP(J),J=1,JMAX)
          WRITE(NOUT,2007)
     2007 FORMAT(1H0,10X,'XLP(J)',/)
          WRITE(NOUT,1007) (J,XLP(J),J=1,JMAX)
C
          READ(NIN,1006) (D(J),J=1,JMAX)
          WRITE(NOUT,2008)
     2008 FORMAT(1H0,10X,'D(J)',/)
          WRITE(NOUT,1007) (J,D(J),J=1,JMAX)
C
          READ(NIN,1006)  (EMO(J),J=1,JMAX)
          WRITE(NOUT,2009)
     2009 FORMAT(1H0,10X,'EMO(J)',/)
          WRITE(NOUT,1007) (J,EMO(J),J=1,JMAX)
C
          READ(NIN,1006) (CONP(J),J=1,JMAX)
          WRITE(NOUT,2010)
     2010 FORMAT(1H0,10X,'CONP(J)',/)
          WRITE(NOUT,1007) (J,CONP(J),J=1,JMAX)
C
          READ(NIN,1006) (CONN(J),J=1,JMAX)
          WRITE(NOUT,2011)
     2011 FORMAT(1H0,10X,'CONN(J)',/)
          WRITE(NOUT,1007) (J,CONN(J),J=1,JMAX)
C
          READ(NIN,1006) (Z(I),I=1,IMAX)
          WRITE(NOUT,2012)
     2012 FORMAT(1H0,10X,'Z(I)',/)
          WRITE(NOUT,1007) (I,Z(I),I=1,IMAX)
C
          READ(NIN,1006) (XM(I),I=1,IMAX)
          WRITE(NOUT,2013)
     2013 FORMAT(1H0,10X,'XM(I)',/)
          WRITE(NOUT,1007) (I,XM(I),I=1,IMAX)
          READ(NIN,1006) (PB(N),N=1,NMAX)
          IF(NMAX)2028,1102,2028
     2028 WRITE(NOUT,2014)
     2014 FORMAT(1H0,10X,'PB(N)',/)
          WRITE(NOUT,1007) (N,PB(N),N=1,NMAX)
          IF(NMAX)1200,1102,1200
     1200 DO 1100 N=1,NMAX
          READ(NIN,1006) (HEAD(M,N),M=1,10)
          WRITE(NOUT,2015)
     2015 FORMAT(1H0,10X,'HEAD(M,N)-N=',I3,/)
```

FORTRAN IV030 SOURCE PROGRAM HYTRAN PROGRAM

```
      WRITE(NOUT,1007) (M,HEAD(M,N),M=1,10)
 1100 CONTINUE
C
      DO 1101   N=1,NMAX
      READ(NIN,1006) (GPM(M,N),M=1,10)
      WRITE(NOUT,2016)
 2016 FORMAT(1H0,10X,'GPM(M,N)-N=',I3,/)
      WRITE(NOUT,1007) (M,GPM(M,N),M=1,10)
 1101 CONTINUE
 1102 CONTINUE
      IF(FLAG1-1)100,457,100
  457 WRITE(NOUT,2053)
 2053 FORMAT('1',14X,(' DEBUGGING PRINTOUT'),/)
  101 WRITE(NOUT,2017)
 2017 FORMAT(1H0,14X, 'IMAX      JMAX       NMAX')
      WRITE(NOUT,1010)IMAX,JMAX,NMAX
 1010 FORMAT(8X,3(6X,I3))
  100 CONTINUE
C
C     ***CHANGE DIMENSIONS OF INPUT DATA TO FT-LB-SEC***
C
      COEFF=(231.*RHO)/(60.*1728.)
      DO 1 J=1,JMAX
      D(J)=D(J)/12.
      AREA(J)=(3.1416*D(J)**2/4.)
      EMO(J)=EMO(J)*COEFF
    1 CONTINUE
      DO 8888 I=1,IMAX
      XM(I)=XM(I)*COEFF
 8888 CONTINUE
      IF(FLAG1-1.)102,103,102
  103 WRITE(NOUT,2018)
 2018 FORMAT(1H0,10X,'D(J)',/)
      WRITE(NOUT,1007) (J,D(J),J=1,JMAX)
      WRITE(NOUT,2019)
 2019 FORMAT(1H0,10X,'AREA(J)',/)
      WRITE(NOUT,1007) (J,AREA(J),J=1,JMAX)
      WRITE(NOUT,2050)
 2050 FORMAT(1H0,10X,'XM(I)',/)
      WRITE(NOUT,1007) (I,XM(I),I=1,IMAX)
      WRITE(NOUT,2062)
 2062 FORMAT(1H0,10X,'EMO(J)',/)
      WRITE(NOUT,1007) (J,EMO(J),J=1,JMAX)
  102 CONTINUE
      IF(NMAX)105,104,105
  105 DO 3 N=1,NMAX
      DO 3 M=1,10
      GPM(M,N)=GPM(M,N)*COEFF
    3 CONTINUE
      IF(FLAG1-1.)104,140,104
```

FORTRAN IV030 SOURCE PROGRAM   HYTRAN   PROGRAM

```
  140 WRITE(NOUT,2016)
      WRITE(NOUT,1007) (M,GPM(M,N),M=1,10)
  104 CONTINUE
C
C     ***FORMATION OF CONNECTION MATRIX AT(K)
      DO 110 K=1,KMAX
      AT(K)=0.
  110 CONTINUE
      L=0
      DO 111 J=1,JMAX
      NP=COMP(J)
      NN=CONN(J)
      NP1=NP+L*IMAX
      NN1=NN+L*IMAX
      L=L+1
      IF(NP1)112,113,112
  112 AT(NP1)=1.
  113 CONTINUE
      IF(NN1)114,115,114
  114 AT(NN1)=-1.
  115 CONTINUE
  111 CONTINUE
      IF(FLAG1-1.)116,117,116
  117 WRITE(NOUT,2029)
 2029 FORMAT(1H0,10X,'AT',/)
      CALL DPRINT(AT,IMAX,JMAX)
  116 CONTINUE
      CALL GMTRA(AT,A,IMAX,JMAX)
      IF(FLAG1-1.)118,119,118
  119 WRITE(NOUT,2030)
 2030 FORMAT(1H0,10X,'A',/)
      CALL DPRINT(A,JMAX,IMAX)
  118 CONTINUE
      CALL GMPRD(A,Z,ZB,JMAX,IMAX,1)
      IF(FLAG1-1.)120,121,120
  121 WRITE(NOUT,2031)
 2031 FORMAT(1H0,10X,'ZB',/)
      WRITE(NOUT,1007) (J,ZB(J),J=1,JMAX)
  120 CONTINUE
      DO 25 J=1,JMAX
      GAMMA(J)=AREA(J)*GRAV*RHO/XL(J)
      GAMMA(J)=GAMMA(J)*DELTAT
   25 CONTINUE
      J=0
      DO 9929 J1=1,KMAX
      J=J+1
      IF(J-JMAX)9928,9928,9927
 9927 J=1
 9928 PROD(J1)=GAMMA(J)*A(J1)
 9929 CONTINUE
```

FORTRAN IV030 SOURCE PROGRAM HYTRAN PROGRAM

```
      IF(FLAG1-1.)2222,2221,2222
 2221 WRITE(NOUT,2052)
 2052 FORMAT(1H0,10X,'PROD(J1)',/)
      CALL DPRINT(PROD,JMAX,IMAX)
 2222 CALL GMPRD(AT,PROD,R,IMAX,JMAX,IMAX)
      IF(FLAG1-1.)122,123,122
  123 WRITE(NOUT,2032)
 2032 FORMAT(1H0,10X,'R',/)
      CALL DPRINT(R,IMAX,IMAX)
  122 CONTINUE
      CALL MINV(R,IMAX,DET,LSPACE,MSPACE)
      CALL GMPRD(R,AT,RR,IMAX,IMAX,JMAX)
      CALL GMPRD(R,XM,RM,IMAX,IMAX,1)
      IF(FLAG1-1.)4444,3333,4444
 3333 WRITE(NOUT,2033)
 2033 FORMAT(1H0,10X,'R',/)
      CALL DPRINT(R,IMAX,IMAX)
  129 WRITE(NOUT,2034)
 2034 FORMAT(1H0,10X,'RR',/)
      CALL DPRINT(RR,IMAX,JMAX)
 151  WRITE(NOUT,2074)
 2074 FORMAT(1H0,10X,'RM')
      WRITE(NOUT,1007) (I,RM(I),I=1,IMAX)
 4444 CONTINUE
      DO 27 J=1,JMAX
      H(J)=0.
   27 CONTINUE
 1110 CONTINUE
      IF(NMAX)142,143,142
  142 DO 28 N=1,NMAX
      J=PB(N)
      GPMX(I)=GPM(I,N)
      HEADX(I)=HEAD(I,N)
      H(J)=SI(GPMX,HEADX,EMO(J),10)
   28 CONTINUE
  143 CONTINUE
      DO 29 J=1,JMAX
      REYN(J)=D(J)*ABS(EMO(J))/(AREA(J)*VISC)
      FRIC(J)=.0055*(1.+(2.*10.**4*ROUGH/D(J)+10.**6/REYN(J))**.333)
      FF(J)=((FRIC(J)*XLP(J)*ABS(EMO(J))*EMO(J))/(D(J)*2.*GRAV*RHO*RHO
     1*(AREA(J)**2)))
      F(J)=FF(J)-ZB(J)
       SUM(J)=GAMMA(J)*(F(J)+H(J))-EMO(J)
   29 CONTINUE
      CALL GMPRD(RR,SUM,RRR,IMAX,JMAX,1)
      CALL GMSUB(RRR,RM,P,IMAX,1)
      CALL GMPRD(A,P,DELP,JMAX,IMAX,1)
      IF(FLAG1-1)451,459,451
  459 IF(TIME-TPR)451,458,458
  458 WRITE(NOUT,2075)TIME
```

```
2075 FORMAT('1',14X,'DEBUGGING PRINTOUT AT TIME=',E14.7,/)
 127 WRITE(NOUT,2021)
2021 FORMAT(1H0,10X,'REYN(J)',/)
     WRITE(NOUT,1007) (J,REYN(J),J=1,JMAX)
     WRITE(NOUT,2022)
2022 FORMAT(1H0,10X,'FRIC(J)',/)
     WRITE(NOUT,1007) (J,FRIC(J),J=1,JMAX)
     WRITE(NOUT,2024)
2024 FORMAT(1H0,10X,'SUM(J)',/)
     WRITE(NOUT,1007) (J,SUM(J),J=1,JMAX)
 444 WRITE(NOUT,2037)
2037 FORMAT(1H0,10X,'DELP,FT',/)
     WRITE(NOUT,1007) (J,DELP(J),J=1,JMAX)
 889 WRITE(NOUT,2035)
2035 FORMAT(1H0,10X,'RRR',/)
     WRITE(NOUT,1007) (I,RRR(I),I=1,IMAX)
451  DO 301 J=1,JMAX
     E(J)=H(J)-DELP(J)
301  EM(J)=EMO(J)-GAMMA(J)*(E(J)+F(J))
     IF(TIME-TPR)136,453,453
453  WRITE(NOUT,2071)TIME
2071 FORMAT('1',14X,'OUTPUT VARIABLES AT TIME=',E14.7,/)
 445 WRITE(NOUT,2026)
2026 FORMAT(1H0,10X,'EM(J),GPM',/)
     DO 2076 J=1,JMAX
     EMG(J)=EM(J)/COEFF
2076 CONTINUE
     WRITE(NOUT,1007) (J,EMG(J),J=1,JMAX)
     WRITE(NOUT,2079)
2079 FORMAT(1H0,10X,'EM(J),LBS/SEC',/)
     WRITE(NOUT,1007) (J,EM(J),J=1,JMAX)
     DO 452 I=1,IMAX
     PP(I)=P(I)/144.
 452 CONTINUE
     WRITE(NOUT,2072)
2072 FORMAT(1H0,10X,'P(I),PSI',/)
     WRITE(NOUT,1007) (I,PP(I), I=1,IMAX)
     WRITE(NOUT,2055)
2055 FORMAT(1H0,10X,'FF(J),FT')
     WRITE(NOUT,1007) (J,FF(J), J=1,JMAX)
     TPR=TPR+DTPR
 136 CONTINUE
     DO 61 J=2,JMAX
     REL(J)=ABS((EM(J)-EMO(J))/DELTAT)
61   CONTINUE
     DO 32 J=2,JMAX
     IF(REL(J)-ERROR)32,32,201
  32 CONTINUE
475  WRITE(NOUT,2071)TIME
     WRITE(NOUT,2026)
```

```
      WRITE(MOUT,1007) (J,EMO(J),J=1,JMAX)
      WRITE(MOUT,2079)
      WRITE(MOUT,1007) (J,EM(J),J=1,JMAX)
      WRITE(MOUT,2072)
      WRITE(MOUT,1007) (I,PP(I), I=1,IMAX)
      WRITE(MOUT,2055)
      WRITE(MOUT,1007) (J,FF(J), J=1,JMAX)
      WRITE(MOUT,2077)
 2077 FORMAT('1',14X,'FINAL STEADY STATE PARAMETERS')
      WRITE(MOUT,2027)
 2027 FORMAT(1H0,10X,'RELATIVE ERROR',/)
      WRITE(MOUT,1007) (J,REL(J),J=2,JMAX)
      WRITE(MOUT,2051)
 2051 FORMAT(1H0,10X,'KOUNT',/)
      WRITE(MOUT,9999)KOUNT
      STOP
  201 IF(KOUNT-NCOUNT)202,203,203
  203 WRITE(MOUT,2057)
 2057 FORMAT(1H0,5X,'THE MAXIMUM ALLOWABLE NO. OF ITERATIONS HAS BEEN
     1REACHED',/)
      STOP
  202 KOUNT=KOUNT+1
      TIME=TIME+DELTAT
      DO 10 J=2,JMAX
      EMO(J)=EM(J)
   10 CONTINUE
      GO TO 1110
      END
```