ANL-76-64

ANL-76-64

PTA-1: A COMPUTER PROGRAM FOR ANALYSIS OF PRESSURE TRANSIENTS IN HYDRAULIC NETWORKS, INCLUDING THE EFFECT OF PIPE PLASTICITY

by

C. K. Youngdahl and C. A. Kot

BASE TECHNOLOGY

U of C-AUA-USERDA

ARGONNE NATIONAL LABORATORY, ARGONNE, ILLINOIS

Prepared for the U. S. ENERGY RESEARCH AND DEVELOPMENT ADMINISTRATION under Contract W-31-109-Eng-38

DISCLAIMER

Printed in the United States of America
Available from
National Technical Information Service
U. S. Department of Commerce
5285 Port Royal Road
Springfield, Virginia 22161
Price: Printed Copy $5.00; Microfiche $3.00

Distribution Category:
LMFBR Structural Materials
and Design Engineering
(UC-79h)

ANL-76-64

ARGONNE NATIONAL LABORATORY
9700 South Cass Avenue
Argonne, Illinois 60439

PTA-1: A COMPUTER PROGRAM FOR ANALYSIS OF PRESSURE TRANSIENTS IN HYDRAULIC NETWORKS, INCLUDING THE EFFECT OF PIPE PLASTICITY

by

C. K. Youngdahl and C. A. Kot

Components Technology Division

November 1976

THIS PAGE

WAS INTENTIONALLY

LEFT BLANK

TABLE OF CONTENTS

	Page
ABSTRACT	7
I. INTRODUCTION	7
II. ASSUMPTIONS	10
III. EQUATIONS	12
A. Characteristic Equations	12
B. Finite-difference Solution along Characteristic Curves	14
C. Determination of Time-Space Grid	15
D. Interpolation in Fixed Time-Space Grid	16
E. Interior-node Calculation	18
F. Junction-node Calculations	18
1. Sudden Expansion or Contraction	19
2. Tee	20
3. Pump	20
4. Acoustic Impedance Discontinuity with No Area Change	20
5. Dummy Junction	21
6. Closed End	21
7. Constant-pressure Boundary	22
8. Far-end Boundary	22
9. Rupture Disk	23
10. Pressure-Source Junction	23
G. Pipe Friction Factor	24
H. Sodium Properties	24
I. Stress-Strain Relations for Piping Materials	25
IV. CODE STRUCTURE	27
A. Main Program	27
1. Specifications	27
2. Input	28
3. Computation of Problem Parameters	28
4. Print Input, Problem Parameters, and Network Arrangement	28
5. Determination of Pipe Connections at Each Junction	28
6. Conversion of Units	29

TABLE OF CONTENTS

Page

7. Initializations at TBEG 29
8. Interior-node Calculation 29
9. Junction-node Calculation 29
10. Print Results at Specified Time Step 30
11. Initializations for Next Time Step 30
12. Printout at End of Problem 30
13. Diagnostic Message Prints 31
14. Input-format Statements 31
15. Output-format Statements 31

B. Subroutine MTPRP 31

C. Subroutine FLPRP(SPWT,DENS,VISC,SNDSPD) 32

D. Subroutine INTRP(L,I,K,U,P,C) 32

E. Subroutine FRCTRM(L,U,G) 32

F. Subroutine AREACH(J) 32

G. Subroutine TEE(J,INCOND,PR) 33

H. Subroutine PUMP(J,LLPMP,HEAD) 33

I. Subroutine IMPED(J) 33

J. Subroutine CLOSED(J) 33

K. Subroutine CONSTP(J,PRES) 33

L. Subroutine FAREND(J) 33

M. Subroutine RUPDSK(J,PRDB,PRDG,KRD) 33

N. Subroutine PRESSO(J,PSO) 34

O. Subroutine PTIME(T,P) 34

V. PTA-1 INPUT REQUIREMENTS 34

VI. DIAGNOSTIC MESSAGES 37

VII. ALTERATIONS TO PROGRAM STORAGE 40

A. Array Sizes in COMMON Statements 41

B. Array Sizes in DIMENSION Statements 41

TABLE OF CONTENTS

Page

APPENDIXES

A. Sample Problem 44

B. Listing of PTA-1. 48

ACKNOWLEDGMENT 83

REFERENCES. 84

LIST OF FIGURES

No.	Title	Page
1.	Stress-Strain Relation	14
2.	Finite-difference Grid for Interior Node	14
3.	Finite-difference Grids for Boundary Nodes. (a) Last Node at Boundary; (b) First Node at Boundary	18
4.	Piping Network for Sample Problem	44
5.	Input Data for Sample Problem	45
6.	Pressure History at Junction 20 and Source Pulse at Junctions 1 and 2	46
7.	Pressure History at Junction 3	46

LIST OF TABLES

No.	Title	Page
I.	Sharing of Data in Numbered Common Blocks	27
II.	Storage Limits	41

PTA-1: A COMPUTER PROGRAM FOR ANALYSIS OF PRESSURE TRANSIENTS IN HYDRAULIC NETWORKS, INCLUDING THE EFFECT OF PIPE PLASTICITY

by

C. K. Youngdahl and C. A. Kot

ABSTRACT

The computer program PTA-1 performs pressure-transient analysis of large piping networks using the one-dimensional method of characteristics applied to a fluid-hammer formulation. The effect of elastic-plastic deformation of piping on pulse propagation is included in the computation. The program is particularly oriented toward the analysis of the effects of a sodium/water reaction on the intermediate heat-transport system of a liquid-metal-cooled fast breeder reactor, but may be applied just as usefully to other pulse sources and other piping systems. PTA-1 is capable of treating complex piping networks and includes a variety of junction types. Pipe friction and nonlinear velocity terms are included in the formulation. The program requires a minimum of input-data preparation and is designed to be easily used and modified. This report contains the governing equations, program structure, input requirements, program listing, and other information for PTA-1.

I. INTRODUCTION

Pressure pulses in the intermediate sodium system of a liquid-metal-cooled fast breeder reactor, such as may originate from a sodium/water reaction in a steam generator, are propagated through the complex sodium piping network to system components such as the pump and intermediate heat exchanger. To assess the effects of such pulses on continued reliable operation of these components and to contribute to system designs which result in the mitigation of these effects, Pressure Transient Analysis (PTA) computer codes are being developed for accurately computing the transmission of pressure pulses through a complicated fluid-transport system, consisting of piping, fittings and junctions, and components.

Pressure pulses resulting from sodium/water reactions may plastically deform the thin-walled piping typically used in LMFBR sodium systems. This plastic deformation has a significant effect on pressure-transient propagation,[1,2] since it limits the peak pressure transmitted out of a pipe to approximately its yield pressure if the pipe is sufficiently long. PTA-1 computes the effect of plastic deformation of the piping on pressure-transient propagation in complex hydraulic networks. Although it was developed for predicting the propagation of pressure pulses in the intermediate-heat-transport system of a sodium-cooled fast breeder reactor, it may also be used to analyze transient propagation in other hydraulic networks for which fluid-hammer theory is appropriate. The effects of cavitation and pipe-support motion are not included in the formulation of PTA-1, but may be incorporated in later codes in the PTA series.

PTA-1 was constructed by combining the complex-piping-system analysis capability of the NATRANSIENT[3,4] program with the modeling of plastic pipe-deformation effects contained in the PLWV[5,6] program, which has very limited system capability. Thus, PTA-1 provides an extension of the well-accepted and verified fluid-hammer formulation[7] for computing hydraulic transients in elastic or rigid piping systems to include plastic-deformation effects. The accuracy of the modeling of these latter effects on transient propagation has been validated[5,6] using results of a Stanford Research Institute experiment.[2]

NATRANSIENT[3,4] applies the one-dimensional method of characteristics to the nonlinear fluid-hammer equations in a fixed time-space grid for each pipe of a flow network. Frictional losses at the pipe walls are computed from the Darcy-Weisback factor, using correlations for friction factors based on local velocity and pipe roughness. The local wave speed is modified for pipe elasticity. NATRANSIENT has good network capability. The variety of junctions and fittings available is important, not only for describing an LMFBR intermediate sodium system, but also for isolating subsystems and constructing models of complicated components such as an intermediate heat exchanger. Assemblage of the flow network and computation of friction factors, local wave speeds, junction losses, and fluid properties are carried out within the program, rather than being required inputs as they are in some other codes. NATRANSIENT's structure is compatible with multidimensional programs being developed under ANL's Large Leak Analysis program for treating fluid response inside the steam generator.

PLWV[5,6] was developed to test a simple computational model for incorporating the effect of elastic-plastic deformation of piping on pressure-transient propagation in a fluid system. The structural interaction model was incorporated into a one-dimensional method-of-characteristics procedure for fluid-hammer analysis. PLWV is limited to a two-pipe system, since this was the arrangement for which experimental data were available.[2] Computed results were found to be in good agreement with these data.

PTA-1 also uses the one-dimensional method of characteristics applied to fluid-hammer analysis of pressure transients in large piping systems. Nonlinear convective terms, pipe friction, fluid compressibility, and wave-speed dependence on pipe deformation are included in the formulation of the governing equations. Various types of junctions and fittings may be specified; these include closed ends, multibranched tees, surge tanks, sudden expansions and contractions, dummy junctions, acoustic-impedance discontinuities, nonreflecting far-end boundaries, and simple models for pumps and rupture disks.

In addition to including the effect of elastic-plastic deformation of piping on transient propagation, PTA-1 differs from NATRANSIENT in being more user-oriented. The anticipated user here is a reactor engineer with some experience in basic FORTRAN programming, but not necessarily expert in the construction and manipulation of large computer codes. Since it is impossible to anticipate all user needs, PTA-1 is structured so as to be easily modified and expanded. Much of the computation is carried out in subroutines that can be replaced or altered, and the main program is divided into subsections that are intended to be readily comprehensible. Programming tricks to minimize the number of instructions or conserve storage have been avoided to preserve clarity and to facilitate alterations to part of the program without disrupting other parts. User input and preliminary computations have been minimized. There are no restrictions on pipe and junction numbering or designation of left and right ends of pipes, and the flow network is assembled by the program. Node spacings, fluid properties, pipe material properties, pipe flow areas, friction factors, wave speeds, and junction losses are computed internally. Some important features of PTA-1 that differ from or are extensions of those in NATRANSIENT follow.

a. The effect of plastic deformation of a pipe is incorporated through a modified fluid wave speed, which varies with deformation history at each computational node in a plastically deforming pipe.

b. Temperature-dependent properties of liquid sodium are computed in a subroutine, which can be replaced if the piping network contains a different fluid.

c. Pipe material properties are computed in a subroutine that can treat six different pipe materials in the same network. Material characterizations contained in the subroutine include temperature-dependent stress-strain relations for Types 304 and 316 stainless steels, Nickel 200 (which has been used in experimental modeling of LMFBR piping[2]), and functional relations that are useful in curve fitting of stress-strain relations.

d. Each type of junction is treated in a separate subroutine to make it easier to substitute improved versions or to add additional types of junctions.

e. The entire pipe-friction loss term, rather than just the friction factor, is computed in a subroutine. Consequently, a different friction-loss formulation can be readily substituted.

f. The source pressure-time relation is computed in a subroutine. In the current version of PTA-1, a table of pressure-time values is input to the program, and the subroutine performs linear interpolation to determine source pressure as a function of time. This subroutine could be replaced by other source models, such as the Zaker-Salmon sodium/water-reaction model[8] used in NATRANSIENT.

g. Pipes and junctions can be numbered arbitrarily; i.e., the numbers need not be consecutive and can be assigned in any order. Consequently, a subsystem of a large network can be analyzed without requiring any renumbering, and pipe and junction designations in the computer output can be maintained as a system is modified.

h. One end of each pipe is designated as the first node end, and the other the last node end, in order to determine a positive coordinate direction for fluid velocity in the pipe. However, these designations are arbitrary in that every junction subroutine can treat any combination of pipe ends. For example, the ends of the two pipes connected at a sudden expansion junction can be both first-node ends, both last-node ends, or one of each.

i. All input data are read into the main program to avoid omissions or misorderings.

j. An improved method of determining node spacing is used to minimize numerical dispersion in the calculation of transient propagation.

k. Many diagnostics are used to determine consistency of problem input and to assist the user in detecting input errors.

l. An attempt has been made to use consistent and reasonably obvious notation throughout the program to facilitate later modification.

Sections II and III contain the assumptions and equations underlying the development of PTA-1, and the structure of the main program and subroutines is described in Sec. IV. A summary list of input data is presented in Sec. V, followed by elaborations on some of the individual input items. Section VI contains explanations of diagnostic messages written by the program when it encounters input inconsistencies or other difficulties. The alterations to COMMON, DIMENSION, and DATA statements needed to reduce or enlarge program storage to conserve core requirements or treat increased system size and complexity are given in Sec. VII. A sample problem and a listing of PTA-1 are contained in the appendixes.

II. ASSUMPTIONS

The standard assumptions underlying one-dimensional fluid-hammer analysis of pressure transients in piping systems are:[7]

a. The axial velocity u is the only nonzero velocity component. This assumption is modified slightly here in that one-dimensional flow is assumed in deriving the governing fluid-motion equations, but fluid movement in the radial direction is accounted for in determining wall-deformation effects.

b. The pressure p and axial velocity u are functions of axial position x and time t only.

c. Changes in fluid density are negligible compared to the density itself. In the governing differential equations, the fluid density is assumed to be a constant and, consequently, independent of position and time, but the bulk compressibility of the fluid is taken into account in computing the wave speed c.

d. Viscous losses in the fluid are neglected.

e. Frictional losses at the pipe wall are included through the Darcy-Weisback friction coefficient f.

The above assumptions all pertain to the treatment of the fluid. To these must be added some assumptions on the influence of pipe-wall deformation on hydraulic-transient propagation. Various modelings of the pipe deformation are possible, ranging from a rigid pipe-wall model with no structure-fluid interaction effects to a detailed modeling of dynamic deformations and stresses in the piping and the resultant interactions of the stress waves and pipe vibrations with the fluid motion. The model used here is essentially the simplest pipe-response model that incorporates some influence of plastic wall deformation on transients in the fluid. It has the advantages of being readily incorporated into standard fluid-hammer analysis procedures and giving results that are conceptually plausible and agree well with available experimental evidence on plastic wall-deformation interaction with fluid-transient propagation.[5,6]

The additional assumptions involved in this pipe response model include:

a. The pipe response is quasi-static; i.e., the pipe deformation is in equilibrium with the fluid-pressure distribution, which varies with x and t. This eliminates all waves traveling through the pipe material.

b. Bending moments in the pipe wall are neglected, and pipe deformations are not required to be continuous functions of x. This implies that the pipe is treated as a series of rings that act independently of each other.

c. Axial stresses and strains in the pipe are neglected.

d. The pipe material is incompressible; this is the usual assumption in plasticity problems.

e. The pipe wall is thin enough that circumferential stress variations across the thickness can be neglected.

f. Circumferential strains are small.

The result of these assumptions on pipe response is that the only influence of pipe deformation on transient propagation in the fluid is through its effect on local wave speed. The wave speed, which is equal to the sound speed in the fluid if wall-deformation effects are neglected, is no longer just a function of fluid properties, but now depends on fluid properties, pipe properties, and pipe-deformation history. Consequently, it can vary with time and position along the pipe, and provision is made in the computational scheme to accommodate this variation.

III. EQUATIONS

The equations governing fluid-hammer analysis of pressure-transient propagation in rigid or elastic piping systems using the one-dimensional method of characteristics are derived in Refs. 3 and 4, standard texts such as Ref. 7, and elsewhere. Modifications to account for the effect of plastic deformation of the piping are derived in Refs. 1, 5, and 6. These derivations will not be repeated here; only the resulting set of governing equations will be summarized.

A. Characteristic Equations

Applying the one-dimensional method of characteristics to fluid flow and continuity relations results in equivalent differential equations that involve only total derivatives with respect to time and apply only along characteristic curves; these are

$$\frac{du}{dt} + \frac{1}{\rho c}\frac{dp}{dt} + g(u) = 0, \tag{1}$$

which holds along the positive characteristic C^+, given by

$$dx = (u + c)dt, \tag{2}$$

and

$$\frac{du}{dt} - \frac{1}{\rho c}\frac{dp}{dt} + g(u) = 0, \tag{3}$$

which holds along the negative characteristic C^-, given by

$$dx = (u - c)dt, \tag{4}$$

where p and u are fluid pressure and velocity at position x and time t, ρ is fluid density, and c is local wave speed. The pipe friction term g(u) is assumed here to be

$$g(u) = \frac{fu|u|}{2D}, \tag{5}$$

where f is the Darcy-Weisback friction factor and D is the pipe diameter.

For a rigid pipe wall, the wave speed is equal to the speed of sound in the fluid and is given by

$$c_0^2 = K/\rho \tag{6}$$

where K is the bulk modulus of the fluid. If the pipe is deforming elastically, the wave speed is given by

$$c^2 = \frac{K/\rho}{1 + \frac{KD}{EH}}, \tag{7}$$

where H is pipe-wall thickness and E is Young's modulus of the pipe material. If portions of the pipe are undergoing plastic deformation, the wave speed is then given by

$$c^2 = \frac{K/\rho}{1 + \frac{KD}{H\left(\frac{d\sigma}{d\varepsilon} - 2\sigma\right)}}, \tag{8}$$

where σ and ε are circumferential stress and strain in the pipe; σ is related to the fluid pressure through

$$\sigma = \frac{pD}{2H}. \tag{9}$$

For rigid or elastic pipe-wall response, the wave speed is a constant for each pipe. On the other hand, the wave speed varies along a plastically deforming pipe, since p and, consequently, σ and $d\sigma/d\varepsilon$ vary with position and time. Moreover, $d\sigma/d\varepsilon$ depends not only on the current value of σ (and p through Eq. 9), but also on prior strain history and the sign of dp. If there has been plastic deformation at a pipe cross section followed by elastic unloading (path 123 in Fig. 1), the yield stress, which was originally σ_1, will be increased by strain hardening to σ_2; stresses such as σ_4, which would have produced plastic deformation originally, will now produce elastic deformation with its correspondingly higher wave speed. If a pipe cross section is deforming plastically (point 2 in Fig. 1), a further pressure increase will produce additional plastic deformation, corresponding to a low wave speed; on the other hand, a pressure decrease will produce elastic unloading corresponding to a higher wave speed.

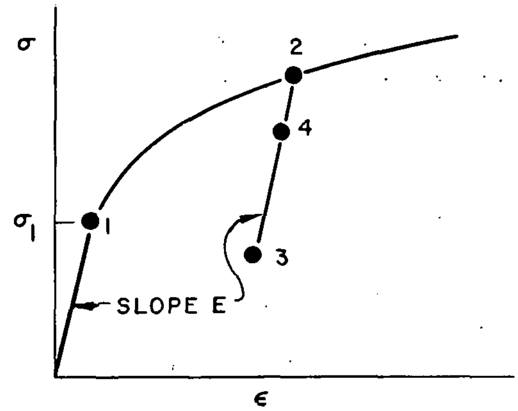


Fig. 1

Stress–Strain Relation. ANL Neg. No. 113–5881.

The maximum stress experienced at each axial node point of a plastically deforming pipe is monitored. If the previous maximum stress at a particular point is less than the original yield stress σ_1, the current stress is compared with σ_1 to determine whether the corresponding deformation is elastic or plastic. If the previous maximum stress is σ_2, which is greater than σ_1, the current stress is compared with σ_2 to determine whether the current deformation is elastic or plastic.

B. Finite-difference Solution along Characteristic Curves

If the solution for pressure and fluid velocity is known at a time t_0 for every position x along a pipe, the solution at a later time $t_0 + \Delta t$ can be found through the relations between du/dt and dp/dt that hold along the characteristic curves. Expressing Eqs. 1 and 3 in finite-difference form for C^+ and C^- characteristics intersecting at point P (Fig. 2) gives

$$u_P - u_A + \frac{1}{\rho c_A^+}(p_P - p_A) + g(u_A^+)\Delta t = 0 \tag{10}$$

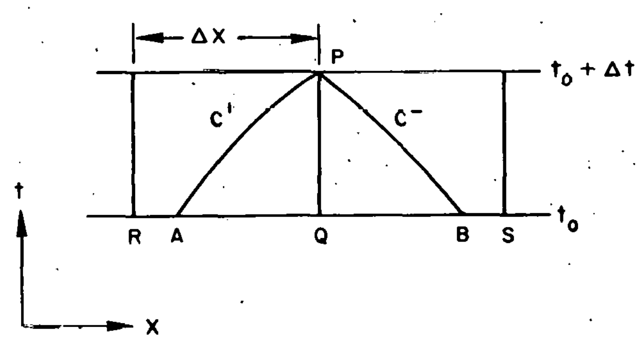
and

$$u_P - u_B - \frac{1}{\rho c_B^-}(p_P - p_B) + g(u_B^-)\Delta t = 0, \tag{11}$$

where c_A^+ and u_A^+ are appropriately averaged values of wave speed and fluid velocity along the C^+ characteristic between points A and P, and c_B^- and u_B^-

Fig. 2

Finite-difference Grid for Interior Node. ANL Neg. No. 113–5882.



are appropriately averaged values of wave speed and fluid velocity along the C^- characteristic between points B and P.

Since the frictional losses at the pipe wall are small, $g(u_A^+)$ and $g(u_B^-)$ are conveniently approximated by

$$\left. \begin{aligned} g(u_A^+) &\approx g_A \\ \text{and} \qquad & \\ g(u_B^-) &\approx g_B \end{aligned} \right\}, \tag{12}$$

where $g_A = g(u_A)$ and $g_B = g(u_B)$ by definition.

If the pipe wall is rigid or is deforming elastically, the wave speed is constant; i.e.,

$$c_A^+ = c_B^- = c, \tag{13}$$

where c is found from Eq. 6 or 7, whichever is appropriate. However, the wave speed can vary significantly along the characteristics if the pipe is deforming plastically; for this case, we will take

$$\left. \begin{aligned} c_A^+ &\approx \tfrac{1}{2}(c_A + c_P) \\ \text{and} \qquad & \\ c_B^- &\approx \tfrac{1}{2}(c_B + c_P) \end{aligned} \right\}, \tag{14}$$

where c_A, c_B, and c_P are the local wave speeds corresponding to conditions at nodes A, B, and P, respectively, and are computed from Eq. 8.

C. Determination of Time-Space Grid

The Courant-Friedrichs-Lewy (CFL)[9] criterion for convergence and stability of the finite-difference scheme used here requires that the time step Δt and axial grid spacing Δx for a pipe satisfy

$$\Delta x \geq (c + |u|)\Delta t. \tag{15}$$

Since the time step is the same for the entire system and the wave speed varies from pipe to pipe if the pipes deform, Δx must be selected for each pipe so as to satisfy the above inequality. For strain-hardening materials, the fluid wave speed corresponding to elastic deformation of the piping is greater than that corresponding to plastic deformation; consequently, the former will be used in determining Δx. We will take

$$\delta_1 \leq \Delta x \leq \delta_2, \tag{16}$$

where

$$\delta_1 = F_1 c\Delta t, \qquad \delta_2 = F_2 c\Delta t. \tag{17}$$

The constant F_1 is chosen to allow for the effect of the fluid speed in the CLF criterion, and F_2 is chosen to prevent excessive numerical dispersion of the transients. In PTA-1, $F_1 = 1.03$ and $F_2 = 1.10$ are used. For a pipe of length L, let n be the closest integer to L/δ_1; i.e.,

$$n = \{L/\delta_1 + 0.5\}, \tag{18}$$

where {} denotes the "greatest-integer" function. We will take

$$\Delta x = L/n, \tag{19}$$

if inequalities 16 are satisfied thereby. If Δx computed from Eq. 19 is less than δ_1, we will take $\Delta x = \delta_1$ and compute a revised pipe length L' from

$$L' = n\delta_1. \tag{20}$$

Similarly, if Δx computed from Eq. 19 is greater than δ_2, we will take $\Delta x = \delta_2$ and compute L' from

$$L' = n\delta_2. \tag{21}$$

If the fluid velocity becomes large enough during a problem computation that violation of the CFL criterion is imminent, the time step is decreased to ensure stability of the solution.

D. Interpolation in Fixed Time-Space Grid

The interpolations required in the fixed time-space grid have two aspects: The locations of the intersections of the characteristics with the constant time line (points A and B of Fig. 2) must be determined; and then values of the desired quantities must be computed at these locations in terms of their values at the grid points. Let v_A^+ be an appropriately averaged value of u + c along the C^+ characteristic through points A and P; v_B^- be an appropriately averaged value of u - c along the C^- characteristic through points B and P; and

$$\theta = \Delta t/\Delta x. \tag{22}$$

Then,

$$\left.\begin{aligned} x_A &= x_Q - v_A^+(x_Q - x_R)\theta, \\ x_B &= x_Q - v_B^-(x_S - x_Q)\theta, \end{aligned}\right\} \tag{23}$$

$$\left.\begin{aligned} p_A &= p_Q - v_A^+(p_Q - p_R)\theta, \\ p_B &= p_Q - v_B^-(p_S - p_Q)\theta, \end{aligned}\right\} \tag{24}$$

and

$$\left.\begin{aligned} u_A &= u_Q - v_A^+(u_Q - u_R)\theta, \\ u_B &= u_Q - v_B^-(u_S - u_Q)\theta. \end{aligned}\right\} \tag{25}$$

For a rigid or elastically deforming pipe, the wave speed is constant along the characteristics and, since the fluid velocity is small compared to the wave speed, we can take

$$v_A^+ \approx u_A + c, \qquad v_B^- \approx u_B - c, \tag{26}$$

where c is computed from Eq. 6 or 7, whichever is appropriate. Simultaneous solution of Eqs. 25 and 26 then gives

$$\left.\begin{aligned} u_A &= \frac{u_Q - c(u_Q - u_R)\theta}{1 + (u_Q - u_R)\theta} \\ &\text{and} \\ u_B &= \frac{u_Q + c(u_S - u_Q)\theta}{1 + (u_S - u_Q)\theta} \end{aligned}\right\}. \tag{27}$$

For plastically deforming pipe, the wave speed varies significantly along the characteristics. For this case, we will take

$$\left.\begin{aligned} v_A^+ &\approx \tfrac{1}{2}(u_A + c_A + u_P + c_P) \\ &\text{and} \\ v_B^- &\approx \tfrac{1}{2}(u_B - c_B + u_P - c_P) \end{aligned}\right\}. \tag{28}$$

Since u_P and c_P (which depends on p_P) are unknown at time t_0, an iterative solution is required between Eqs. 8, 24, 25, and 28 and equations for u_P and p_P given in subsequent parts of this section. In the computation of c_A and c_B, the maximum pressures experienced at points A and B must be known up to time t_0; let $(p_{max})_A$ and $(p_{max})_B$ be these values. Then, using linear interpolation, we obtain

$$\left.\begin{aligned}(p_{max})_A &= (p_{max})_Q - v_A^+[(p_{max})_Q - (p_{max})_R]\theta \\ &\text{and} \\ (p_{max})_B &= (p_{max})_Q - v_B^-[(p_{max})_S - (p_{max})_Q]\theta\end{aligned}\right\}, \qquad (29)$$

where $(p_{max})_Q$, etc., are stored values at the node points. If $p_A < (p_{max})_A$, the local deformation at point A is elastic and c_A is computed from Eq. 7; if $p_A \geq (p_{max})_A$ and p_A also exceeds the pipe yield pressure, the deformation is plastic and c_A is computed from Eq. 8 and the stress-strain relation. An analogous procedure is used to determine c_B.

E. Interior-node Calculation

At an interior node, Eqs. 10 and 11 can be solved for u_P and p_P to give, using Eqs. 12,

$$\left.\begin{aligned}u_P &= [c_A^+(u_A - g_A\Delta t) + c_B^-(u_B - g_Bt) + (p_A - p_B)/\rho]/(c_A^+ + c_B^-) \\ &\text{and} \\ p_P &= [p_A/c_A^+ + p_B/c_B^- + \rho(u_A - g_A\Delta t - u_B + g_B\Delta t)]c_A^+c_B^-/(c_A^+ + c_B^-)\end{aligned}\right\}. \qquad (30)$$

For rigid or elastic pipe walls, these equations give explicit closed-form relations for u_P and p_P. For plastically deforming pipe walls, c_A^+, c_B^-, and, consequently, the interpolations for p_A, p_B, u_A, and u_B depend on u_P and p_P. Initial trial values for u_P, p_P, and the interpolated quantities are assumed; c_A, c_B, and c_P are determined from Eq. 8; v_A^+ and v_B^- are determined from Eqs. 28; and p_A, p_B, u_A, u_B, $(p_{max})_A$, and $(p_{max})_B$ are computed from Eqs. 24, 25, and 29. New values of c_A^+ and c_B^- are then determined and the procedure repeated until the values of the interpolated quantities converge. New values of u_P and p_P are then computed from Eqs. 30, and the iterations are continued until they converge.

F. Junction-node Calculations

Typical finite-differences grids for boundary nodes are shown in Fig. 3, where Fig. 3a indicates a last-node pipe end and Fig. 3b a first-node

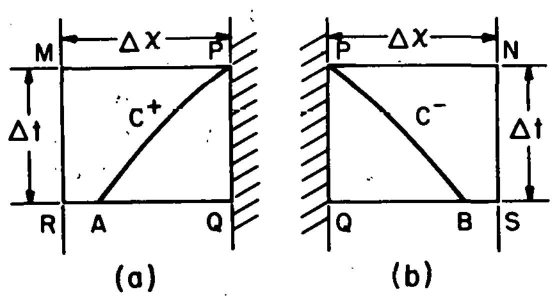


Fig. 3

Finite-difference Grids for Boundary Nodes. (a) Last Node at Boundary; (b) First Node at Boundary.

end. Equations for computing the fluid velocity and pressure at the various types of junctions incorporated as subroutines in PTA-1 follow.

1. Sudden Expansion or Contraction

Assume the area-change junction connects the last-node end of pipe LN, which has area $(A)_{LN}$, to the first-node end of pipe L1, which has area $(A)_{L1}$. The other combinations of pipe-end connections are covered in the subroutine, but their equations will not be listed here since they are easily deduced from the given case. From Ref. 4 and Fig. 3,

$$\left.\begin{aligned}
(u_P)_{LN} &= \frac{\gamma}{\beta + \sqrt{\beta^2 - \gamma\delta}}, \\
(u_P)_{L1} &= R(u_P)_{LN}, \\
(p_P)_{LN} &= p_A - \rho c_A^+[(u_P)_{LN} - u_A + g_A \Delta t], \\
\text{and} \quad (p_P)_{L1} &= p_B + \rho c_B^-[(u_P)_{L1} - u_B + g_B \Delta t],
\end{aligned}\right\} \quad (31)$$

where p_A, c_A^+, u_A, and g_A pertain to pipe LN, and p_B, c_B^-, and u_B, and g_B pertain to pipe L1. In the above,

$$\left.\begin{aligned}
R &= (A)_{LN}/(A)_{L1}, \\
\beta &= c_A^+ + Rc_B^-, \\
\gamma &= 2[c_A^+(u_A - g_A \Delta t) + c_B^-(u_B - g_B \Delta t) + (p_A - p_B)/\rho], \\
\text{and} \quad \delta &= 1 - R^2 - K.
\end{aligned}\right\} \quad (32)$$

The loss coefficient K is given by[10]

$$\left.\begin{aligned}
K &= (1 - R)^2, \text{ if } R < 1 \text{ and } U > 0, \\
K &= -(1 - R)^2, \text{ if } R > 1 \text{ and } U < 0, \\
K &= 0.45R(R - 1), \text{ if } R > 1 \text{ and } U > 0, \\
K &= -0.45(1 - R), \text{ if } R < 1 \text{ and } U < 0, \\
K &= 0, \text{ if } R = 1 \text{ or } U = 0,
\end{aligned}\right\} \quad (33)$$

where

$$U = \tfrac{1}{2}[R(u_R)_{LN} + (u_S)_{L1}]. \tag{34}$$

2. Tee

Consider a tee junction connecting an arbitrary* number of pipes, some of which may be connected at their first-node ends and the remainder at their last-node ends. The pressure is the same for all pipe ends connected at the tee; it is given by

$$p_P = \left\{ \sum_{LN} A_{LN}[\rho(u_A - g_A\Delta t) + p_A/c_A^+]_{LN} + \sum_{L1} A_{L1}[-\rho(u_B - g_B\Delta t) + p_B/c_B^-]_{L1} \right\} \Big/ \left\{ \sum_{LN} (A/c_A^+)_{LN} + \sum_{L1} (A/c_B^-)_{L1} \right\}, \tag{35}$$

where $\sum_{LN}$ denotes summation over all last-node pipe ends and $\sum_{L1}$ denotes summation over all first-node pipe ends. The fluid velocities at a typical last-node end and first-node end are

$$\left.\begin{aligned} (u_P)_{LN} &= [u_A - g_A\Delta t - (p_P - p_A)/(\rho c_A^+)]_{LN} \\ \text{and} \quad (u_P)_{L1} &= [u_B - g_B\Delta t - (p_P - p_B)/(\rho c_B^-)]_{L1} \end{aligned}\right\}. \tag{36}$$

3. Pump

The simple pump model used here treats the pump as a tee junction, with the pump head added to the pressure at the pump end of the pipe representing the outlet of the pump.

4. Acoustic Impedance Discontinuity with No Area Change

Consider two pipes with the same flow area but differing acoustic impedances because of different wall thicknesses or material properties. Let the last-node end of one pipe, designated LN, be connected to the first-node end of the other pipe, designated L1, at a junction. Then

*PTA-1, like NATRANSIENT, currently handles up to six pipes connected at a tee. This can be easily extended by increasing storage designations if more branches are needed in modeling a complex component.

$$\left.\begin{aligned}(u_P)_{L1} &= (u_P)_{LN} \\ &= [c_A^+(u_A - g_A\Delta t) + c_B^-(u_B - g_B\Delta t) + (p_A - p_B)/\rho]/(c_A^+ + c_B^-) \\ \text{and} & \\ (p_P)_{L1} &= (p_P)_{L1} \\ &= [p_A/c_A^+ + p_B/c_B^- + \rho(u_A - g_A\Delta t - u_B + g_B\Delta t)]c_A^+c_B^-/(c_A^+ + c_B^-)\end{aligned}\right\}. \quad (37)$$

These equations are the same as those of Eq. 30 for an interior node, except that here the set u_A, p_A, g_A, and c_A^+ and the set u_B, p_B, g_B, and c_B^- refer to different pipes. The other pipe-end combinations, i.e., first-node end connected to first-node end and last-node end connected to last-node end, are also treated in the subroutine.

Note that the sudden expansion or contraction case with R = 1 is equivalent to the acoustic-impedance discontinuity case, i.e., Eqs. 31-33 with R = 1 reduce to Eq. 37. Consequently, both cases could easily be computed with the same subroutine. This is not done in PTA-1 for two reasons: First, using the sudden-area-change equations to compute the acoustic-impedance discontinuity case is more awkward and time-consuming than using Eq. 37 directly. The second, and more important reason, is that it may be desirable to revise the modeling of one or both of these junction types in subsequent versions of the program; it will be easier to do this if the cases are treated separately.

5. Dummy Junction

For systems having many short pipes and a few long pipes, input of the desired time step may result in violation of the limit on maximum number of computational nodes per pipe. Rather than raise this limit and thereby increase core-storage requirements, it may be expedient to break the long pipes into two or more pipes by inserting dummy junctions. A dummy junction may also be useful for reserving a location for insertion of a tee that connects to a subsystem whose effect on the main system will be determined later, or for identifying the location of a pressure transducer.

The computations for u_P and p_P at the dummy junction are identical to those at the acoustic impedance discontinuity and are performed by the same subroutine.

6. Closed End

For a pipe with its first-node end connected to a closed-end junction,

$$\left.\begin{aligned} u_P &= 0 \\ &\text{and} \\ p_P &= p_B - \rho c_B^-(u_B - g_B \Delta t) \end{aligned}\right\} . \tag{38}$$

If the last-node end of a pipe is closed,

$$\left.\begin{aligned} u_P &= 0 \\ &\text{and} \\ p_P &= p_A + \rho c_A^+(u_A - g_A \Delta t) \end{aligned}\right\} . \tag{39}$$

7. Constant-pressure Boundary

For a pipe with its first-node end connected to a constant-pressure reservoir at pressure p_{RES},

$$\left.\begin{aligned} p_P &= p_{RES} \\ &\text{and} \\ u_P &= u_B - g_B \Delta t + (p_{RES} - p_B)/(\rho c_B^-) \end{aligned}\right\} . \tag{40}$$

It its last-node end is connected to the junction,

$$\left.\begin{aligned} p_P &= p_{RES} \\ &\text{and} \\ u_P &= u_A - g_A \Delta t - (p_{RES} - p_A)/(\rho c_A^+) \end{aligned}\right\} . \tag{41}$$

Numerical studies indicate that the effect of gas compressibility in a surge tank with a gas space on fluid-transient propagation in piping systems of the type considered here is small. Consequently, surge tanks can be modeled as constant-pressure boundaries.

8. Far-end Boundary

In the analysis of subsystems of a complex network, it is useful to have a boundary that transmits pressure waves out of the subsystem without reflecting them. This is easily accomplished by putting the fluid velocity and pressure at the far-end junction equal to their values at the adjacent node pipe; i.e., if the first-node end of a pipe is connected to a far-end junction, then (see Fig. 3)

$$p_P = p_N, \quad u_P = u_N, \tag{42}$$

and if the last-node end is connected, then

$$p_P = p_M, \quad u_P = u_M. \tag{43}$$

9. Rupture Disk

The simple rupture-disk model used here is to treat it as a closed-end boundary until the burst pressure p_{RDB} of the disk is attained and to treat it thereafter as a constant-pressure boundary at pressure p_{RDG}. Let t_{RD} be the time at which the pressure at the disk first reaches p_{RDB}; then, for the first-node end of a pipe connected to the rupture disk,

$$\left.\begin{array}{l} \left.\begin{array}{l} u_P = 0, \\ p_P = p_B - \rho c_B^-(u_B - g_B \Delta t), \end{array}\right\} t \le t_{RD}\ (p_P \le p_{RDB}) \\ \left.\begin{array}{l} p_P = p_{RDG}, \\ u_P = u_B - g_B \Delta t + (p_{RDG} - p_B)/(\rho c_B^-). \end{array}\right\} t > t_{RD} \end{array}\right\} \tag{44}$$

If the last-node end is connected to the disk,

$$\left.\begin{array}{l} \left.\begin{array}{l} u_P = 0, \\ p_P = p_A + \rho c_A^+(u_A - g_A \Delta t), \end{array}\right\} t \le t_{RD}\ (p_P \le p_{RDB}) \\ \left.\begin{array}{l} p_P = p_{RDG}, \\ u_P = u_A - g_A \Delta t - (p_{RDG} - p_A)/(\rho c_A^+). \end{array}\right\} t > t_{RD} \end{array}\right\} \tag{45}$$

10. Pressure-Source Junction

The pressure-source junction is similar to a constant-pressure boundary, except that the pressure $p_{SO}(t)$ at the junction is time-dependent and is obtained from the pressure-source subroutine. Up to six pipes may be connected at a source junction; this number can be increased by changing storage allotments or by designating several pressure-source junctions, all of which will experience the same pressure-history input.

For each first-node pipe end connected to a source junction,

$$\left.\begin{aligned} p_P &= p_{SO}(t) \\ &\text{and} \\ u_P &= u_B - g_B\Delta t + (p_{SO} - p_B)/(\rho c_B^-) \end{aligned}\right\}. \tag{46}$$

and for each last-node end,

$$\left.\begin{aligned} p_P &= p_{SO}(t) \\ &\text{and} \\ u_P &= u_A - g_A\Delta t - (p_{SO} - p_A)/(\rho c_A^+) \end{aligned}\right\}. \tag{47}$$

G. Pipe Friction Factor

Frictional losses at the pipe wall are calculated using Eq. 5. The friction factor f is found from the Colebrook-White correlation

$$\left.\begin{aligned} f &= \frac{64}{N_{Re}}, \quad N_{Re} \le 3000, \\ &\text{and} \\ \frac{1}{\sqrt{f}} &= 1.14 - 2\log_{10}\left(\frac{R_f}{D} + \frac{9.35}{N_{Re}\sqrt{f}}\right), \quad N_{Re} > 3000, \end{aligned}\right\} \tag{48}$$

where R_f is the pipe roughness and N_{Re} is the Reynolds number, defined by

$$N_{Re} = \rho u D/\mu, \tag{49}$$

with μ being the dynamic viscosity of the fluid. The second of Eqs. 48 requires an iterative solution; initial estimates for f are given by

$$\left.\begin{aligned} f &\approx 0.316/N_{Re}^{1/4} \quad \text{for} \quad R_f/D \le 10^{-4} \\ &\text{and} \\ f &\approx [1.14 - 2\log_{10}(R_f/D)]^{-2} \quad \text{for} \quad R_f/D > 10^{-4} \end{aligned}\right\}. \tag{50}$$

H. Sodium Properties

Temperature-dependent properties of liquid sodium are computed from correlations recommended by Golden and Tokar.[11]

The specific weight γ of sodium in lb/ft^3, calculated from Eq. 2.1 of Ref. 11, is

$$\gamma = 59.566 - 7.9504 \times 10^{-3}T - 0.2872 \times 10^{-6}T^2 + 0.06035 \times 10^{-9}T^3,$$

$$208°F \le T \le 2500°F, \quad (51)$$

and the corresponding density ρ in lb-sec^2/ft^4 is

$$\rho = \gamma/g, \quad (52)$$

where $g = 32.2$ ft/sec^2 is the acceleration of gravity.

The dynamic viscosity μ in lb-sec/ft^2, using Eq. 5.19a of Ref. 11, is calculated from

$$\mu = \{\exp[2.303(1.0203 + 397.17/(T + 460)$$

$$-0.4925 \log_{10}(T + 460))]\}/3600/32.2, \quad (53)$$

where T is in degrees Fahrenheit.

Golden and Tokar recommend a linear dependence of sound speed on temperature. Based on tabulated values in their Appendix E, c_0 in ft/sec is calculated from

$$c_0 = 8285 - 2187(T - 210)/2290. \quad (54)$$

I. Stress-Strain Relations for Piping Materials

The temperature-dependent stress-strain relations for Types 304 and 316 stainless steels were obtained from Ref. 12, which gives equations for fitted curves.

Reference 2 provides a stress-strain curve for Nickel 200. (This material is useful for experimental simulation of reactor piping since its stress-strain characteristics at room temperature are similar to those for stainless steels at reactor temperatures.) This curve was fitted by linear elastic and plastic regions connected by a plastic region that is part of an ellipse. This may be a useful functional form for fitting other material data; it is given by

$$\left.\begin{array}{l} \sigma = E\epsilon, \quad 0 \le \epsilon \le \epsilon_1 \text{ (linear elastic)}; \\ C_1(\sigma/\sigma_0)^2 + C_2(\sigma/\sigma_0) + C_3(\epsilon/\epsilon_0) + C_4(\epsilon/\epsilon_0)^2 + C_5 = 0, \\ \quad \epsilon_1 < \epsilon < \epsilon_2 \text{ (elliptic plastic)}; \\ \sigma = \sigma_0 + E_p(\epsilon - \epsilon_0), \quad \epsilon_2 \le \epsilon \text{ (linear plastic)}; \end{array}\right\} \quad (55)$$

where σ and ϵ are stress and strain, E and E_p are the elastic modulus and slope of the linear plastic region, and the point (ϵ_0, σ_0) is the intersection of the extended linear elastic and plastic regions. The constants C_i are chosen so that the curves and their slopes are continuous at the points (ϵ_1, σ_1) and (ϵ_2, σ_2); they are given by

$$\left.\begin{aligned}
C_1 &= (\gamma_2 - 1)^2 - R_m^2(1 - \gamma_1)^2, \\
C_2 &= -2[\gamma_1(\gamma_2 - 1)^2 - R_m(\gamma_2 - \gamma_1)(\gamma_2 - 1)(1 - \gamma_1) - R_m^2\gamma_2(1 - \gamma_1)^2], \\
C_3 &= -2R_m(\gamma_2 - \gamma_1)[(\gamma_2 - 1)(1 - \gamma_1) + R_m(1 - \gamma_1\gamma_2)], \\
C_4 &= R_m^2(\gamma_2 - \gamma_1)(2 - \gamma_1 - \gamma_2), \\
&\text{and} \\
C_5 &= (1 - R_m^2)\gamma_1^2(\gamma_2 - 1)^2,
\end{aligned}\right\} \quad (56)$$

where

$$R_m = E_p/E, \quad \gamma_1 = \sigma_1/\sigma_0, \quad \gamma_2 = \sigma_2/\sigma_0. \quad (57)$$

An essentially exact fit to the Nickel 200 data given in Ref. 2 is obtained by taking $E = 30 \times 10^6$ psi (207×10^6 kPa), $\sigma_0 = 29{,}000$ psi (2×10^5 kPa), $R_m = 0.0135$, $\gamma_1 = 3/4$, and $\gamma_2 = 5/4$.

IV. CODE STRUCTURE

The PTA-1 code consists of the MAIN program; the MTPRP subroutine for computing piping material properties and associated fluid wave speeds; the FLPRP subroutine for computing fluid properties; the INTRP subroutine for performing interpolations in the time-space grid; the FRCTRM subroutine for computing the frictional loss term; the nine junction subroutines AREACH, TEE, PUMP, IMPED, CLOSED, CONSTP, FAREND, RUPDSK, and PRESSO; and the pulse-source subroutine PTIME.

Throughout the program, pipes are referred to by either LPIPE(L) or L, where LPIPE is the user-supplied identification number and L is the corresponding program-generated identification number. Similarly, junctions are referred to by either JUN(J) or J, where JUN is the user-supplied identification number and J is the corresponding program-generated identification number. The user-supplied numbers are arbitrary, while L = 1, 2, 3, ..., NPIPE, and J = 1, 2, 3, ..., NOJUN, where NPIPE and NOJUN are the numbers of pipes and junctions in the system, respectively.

A. Main Program

The main program receives all input data and checks its internal consistency, assembles the piping-network model, performs manipulations and initializations needed to start the method of characteristics computations, performs the interior-node computations, calls the appropriate junction subroutines, prints output, and resets variables for the next time increment. It is composed of the subsections described below.

1. Specifications

This subsection consists of COMMON, DIMENSION, and DATA statements. The sharings of numbered common blocks (R1-R5) with subroutines are indicated in Table I. Alterations to specification statements needed to alter storage requirements or network limitations are discussed in Sec. VII.

TABLE I. Sharing of Data in Numbered Common Blocks

Routine	Common Block R1	R2	R3	R4	R5	Routine	Common Block R1	R2	R3	R4	R5
MAIN	X	X	X	X	X	IMPED	X				
MTPRP	X		X			CLOSED	X				
FLPRP			X			CONSTP	X				
INTRP	X			X		FAREND	X				
FRCTRM	X	X				RUPDSK	X				
AREACH	X					PRESSO	X				
TEE	X					PTIME					X
PUMP	X										

2. Input

All input to the PTA-1 code is accomplished in this subsection, and some input diagnostics are performed. Input data requirements are listed in Sec. V, and diagnostic messages are explained in Sec. VI.

3. Computation of Problem Parameters

The fluid-properties subroutine FLPRP is called to compute the fluid density, dynamic viscosity, and sound speed at system temperature.

Pipe friction is based on pipe diameter D, and expansion loss coefficients are based on flow area A. In modeling a component as an arrangement of equivalent pipes, it may be desirable to have $A \neq \pi D^2/4$ for some of these equivalent pipes; D and A can then be prescribed independently. If $A = \pi D^2/4$, however, the input for A can be left blank and A will be computed from the diameter; this computation is performed in this subsection.

The material-properties subroutine MTPRP is called to compute yield pressure, fluid-wave speed corresponding to elastic deformation, and preliminary material information for each pipe in the system.

The axial grid spacing for each pipe is determined from Eqs. 16-21 for the input time step.

4. Print Input, Problem Parameters, and Network Arrangement

The input data from subsection 2, computed problem parameters from subsection 3, and pipe-network-arrangement information, such as junction identification numbers and types associated with each pipe, are printed to provide input verification and a record of the problem statement.

5. Determination of Pipe Connections at Each Junction

The junction identification numbers for the ends of each pipe are input data that specify the network arrangement. The inverse information is determined in this subsection; i.e., the pipes connected to each junction are determined in order to facilitate subsequent junction-node computations. The quantity ML1(J) is the number of first-node pipe ends connected at junction number JUN(J), and MLN(J) is the number of last-node ends connected there. The quantity LJ1(J,M), $0 \leq M \leq$ ML1(J), implies that pipe number LPIPE(LJ1) is connected at its first-node end to junction number JUN(J); similarly, LJN(J,M), $0 \leq M \leq$ MLN(J), indicates that pipe number LPIPE(LJN) is connected at its last node to junction number JUN(J). The lists ML1, MLN, LJ1, and LJN are passed to the junction subroutines through COMMON. A number of diagnostic checks are provided in this subsection, such as verifying that each two-pipe junction has two pipes connected to it.

6. Conversion of Units

Input to the problem is in customary units, e.g., pipe diameters in inches and pipe lengths in feet. In this subsection, all quantities are converted to the foot/pound-force/second system of units.

7. Initializations at TBEG

The initializations needed to start the method of characteristics computation for the pipe network at time TBEG are performed in this subsection. These include initializing the pressure and velocity matrices P(L,I,K) and U(L,I,K), whose elements give the pressure and velocity at node I of pipe number LPIPE(L) at time t_0 (K = 1) and $t_0 + \Delta t$ (K = 2). Another quantity that is initialized is KPLAS(L). If KPLAS(L) = 0, then pipe number LPIPE(L) has not deformed plastically; if KPLAS(L) = 1, then pipe number LPIPE(L) is the first pipe to deform plastically; if KPLAS(L) = 2, then pipe number LPIPE(L) is the second pipe to deform plastically; etc. Tests of whether KPLAS(L) is greater than or equal to zero provide a convenient means of determining whether an iterative solution is required, and positive values of KPLAS provide convenient indices for storing data pertaining to plastically deforming pipes.

8. Interior-node Calculation

The fluid velocity and pressure at the interior nodes of each pipe are computed in this subsection for a time step Δt, using Eqs. 30. A once-through computation is used for each elastically deforming or rigid pipe, and and iterative procedure is followed for each plastically deforming pipe. The INTRP subroutine is called to perform interpolations in the time-space grid, and FRCTRM is called to compute the frictional loss term.

If the elastic-plastic boundary is close to a computational point and the stress-strain curve has a kink at the yield point, the numerical solution may oscillate rather than converge. An averaging technique is then used to break up the oscillatory pattern and encourage convergence of the iterative procedure.

If the computed pressure somewhere in a pipe that has not previously deformed plastically exceeds its yield pressure, the pipe is designated as a plastically deforming pipe and its pressure and velocity distributions are recomputed for the time step. A message is written denoting the inception of plastic deformation for the pipe.

9. Junction-node Calculation

The fluid velocity and pressure at each junction node are computed by calling the appropriate junction subroutine. Iteration is used if one or more of the pipes connected at the junction have undergone plastic deformation. An averaging technique similar to that employed in the previous subsection is used to disrupt nonconvergent numerical oscillations around the solution. The initiation of plastic deformation at a junction is detected and an appropriate message written. Similarly, an announcement is printed if a rupture disk bursts during the time step.

Some diagnostic checks are performed. For example, if a junction is specified as a rupture disk, but no disk data are input for that junction, an error message is written and the computation is terminated.

10. Print Results at Specified Time Step

The results of the interior- and junction-node computations are printed out in this subsection. Input data specify frequency of output (every time step, every other time step, every third time step, etc.), whether results for all pipes or selected pipes are printed, and printout detail (every node in the selected pipes, every other node, etc.)

Typical output at the end of a time step consists of the time and the source pressure at that time, followed by arrays giving location, pressure, velocity, and sound speed at the specified nodes of the specified pipes. Each array is identified by pipe number and the numbers of the junctions that the pipe connects. The positive velocity direction in a pipe is from the first node to the last node. Pressures are converted to psi before printing.

11. Initializations for Next Time Step

Various initializations and updatings are performed in this subsection to prepare for the next time step. An updated record is maintained of the maximum pressure experienced by each pipe and the time at which it occurs; this information is printed at the end of the problem. The numerical-stability criterion is checked, and Δt is shortened if necessary (see Sec. III.C). The record of maximum pressures attained at each node in a plastically deforming pipe is updated for use in incorporating history effects into the wave-speed computation (see Sec. III.A). The time is incremented by Δt, and the new source pressure is computed. The arrays P(L,I,1) and U(L,I,1) are replaced by corresponding elements from P(L,I,2) and U(L,I,2).

12. Printout at End of Problem

After results of the final time step are printed, some additional information is printed by this subsection at the end of the problem. This includes (1) the maximum pressure experienced by each pipe and the time at which it occurred, (2) the identification numbers of the pipes that have deformed plastically, and (3) identification of the rupture disks that have burst and the corresponding times.

13. Diagnostic Message Prints

PTA-1 has many diagnostic messages to identify input inconsistencies and errors and to indicate computational difficulties. These messages are contained in this subsection of the program and are described in detail in Sec. VI.

14. Input-format Statements

Formats for input data are contained in this subsection.

15. Output-format Statements

Output formats, except for diagnostic messages, are contained in this subsection.

B. Subroutine MTPRP

The MTPRP subroutine computes fluid wave speed as a function of pressure for various piping materials with elastic-plastic deformation effects included. The version in PTA-1 has provision for treating six different piping materials, which are identified by a material number MAT. Currently, MAT = 1 is Type 304 stainless steel; MAT = 2 is Type 316 stainless steel; MAT = 3 is the functional form for fitting stress-strain curves described in Eqs. 55-57, with the material constants for Nickel 200; MAT = 4 and MAT = 5 are bilinear stress-strain relations; and MAT = 6 is a rigid material. System temperature and fluid properties, such as sound speed and density, are available through COMMON. There are three entry points in the MTPRP subroutine:

1. Entry ELPRP(MAT, HD, PYLD, CELAS) is called from the "Computation of Problem Parameters" subsection of the main program. The yield pressure PYLD and elastic wave speed CELAS are computed for a pipe having a thickness-to-diameter ratio HD and made of material type MAT. PYLD is related to the yield stress of the material through Eq. 9, and CELAS is computed using Eq. 7. Various parameters, such as elastic modulus, yield stress, and the C_i of Eqs. 56, are determined for each material the first time the material type is called; if KPR(MAT) = 0, the material type MAT has not been used in some previous call of ELPRP, and if KPR(MAT) = 1, it has been used.

2. Entry WRMAT(MAT) is called from the "Print Input, Problem Parameters, and Network Arrangement" subsection of the main program. It prints a message describing material type MAT. The logic in the main program is such that a material message is printed only once, no mater how many pipes are made of the material, and messages for only the materials used in the system are printed.

3. Entry WVSPD(MAT, HD, P, PX, C) is called to calculate the fluid-wave speed C as a function of pressure P, using Eqs. 8 and 9, at a point in a pipe where the previous maximum pressure is PX. The pipe is made of material MAT and has a thickness-to-diameter ratio HD.

C. Subroutine FLPRP(SPWT, DENS, VISC, SNDSPD)

The FLPRP subroutine computes the specific weight (SPWT), density (DENS), dynamic viscocity (VISC), and sound speed (SNDSPD) of the fluid contained in the piping system. The system temperature, which is a problem input, is available from the main program through COMMON. The current version of FLPRP used in PTA-1 computes liquid-sodium properties from Eqs. 51-54.

D. Subroutine INTRP(L,I,K,U,P,C)

The INTRP subroutine performs the interpolations described in Sec. III.D. If K = 1, the interpolation is along the C^+ characteristic, and the calculated values of fluid velocity U, pressure P, and wave speed C correspond to u_A, p_A, and c_A^+, respectively. If K = 2, the interpolation is along the C^- characteristic, and the returned values of U, P, and C correspond to u_B, p_B, and c_B^-, respectively. (See Figs. 2 and 3 and Eqs. 13, 14, and 22-29.) Grid point Q of Figs. 2 and 3 corresponds to node I of pipe number LPIPE(L).

If KPLAS(L) = 0, the pipe has not undergone plastic deformation and Eqs. 24, 26, and 27 are used. If KPLAS(L) > 0, the pipe has had previous plastic deformation, and Eqs. 24, 25, 28 and 29 are used in an iterative technique. If MPX = 0, the value of the wave speed c_P in Eqs. 28 is computed from conditions at point P. If MPX > 0, the iterative solution at point P performed in the main program is converging poorly and an averaged value of c_P is then passed from the main program to INTRP through COMMON.

E. Subroutine FRCTRM(L,U,G)

The friction term G, defined in Eq. 5, is computed for fluid velocity U in pipe number LPIPE(L). Fluid properties, pipe roughness, and pipe diameter are passed through COMMON. The pipe friction factor f is computed as described in Sec. III.G. If KFRIC = 0, frictional losses are neglected and the friction term is set to zero.

F. Subroutine AREACH(J)

The AREACH subroutine performs the boundary-node calculation at the area change (sudden expansion or contraction) at junction number JUN(J), as described in Sec. III.F.1. The basic case treated is a last-node end connected to a first-node end. The other two cases, i.e., two first-node ends connected or two last-node ends connected, are converted first to equivalent basic cases and then converted back at the end of the calculation.

G. Subroutine TEE(J,INCOND,PR)

The TEE subroutine computes the pressure and fluid velocities at the tee at junction number JUN(J) according to the procedure described in Sec. III.F.2.

H. Subroutine PUMP(J,LLPMP,HEAD)

The PUMP subroutine computes the pressures and fluid velocities at the pump at junction number JUN(J), as described in Sec. III.F.3. The pump head is HEAD, and its outlet is into pipe number LPIPE(LLPMP). The pump is modeled as a tee, with the pump head being subtracted in performing the flow balance at the tee junction. The head is then added on again in computing the pressure at the outlet pipe.

I. Subroutine IMPED(J)

The boundary-node calculation at the dummy junction or acoustic-impedance discontinuity at junction number JUN(J) is performed by the IMPED subroutine, using the procedures described in Sec. III.F.4 and III.F.5. As in the AREACH subroutine, the last-node end of one pipe connected to the first-node end of another is treated as the basic case. The other possibilities are converted first to equivalent basic cases and then converted back at the end of the calculation.

J. Subroutine CLOSED(J)

The CLOSED subroutine performs the boundary-node calculation at the closed end located at junction number JUN(J), using Eqs. 38 and 39. It treats either a first-node pipe end or a last-node end connected to the junction.

K. Subroutine CONSTP(J,PRES)

The CONSTP subroutine performs the boundary-node calculation at the constant-pressure reservoir at pressure PRES located at junction number JUN(J), using Eqs. 40 and 41. Either a first-node pipe end or a last-node end can be connected to the junction.

L. Subroutine FAREND(J)

The far-end boundary condition at junction JUN(J) is calculated as described in Sec. III.F.8. Either a first-node pipe end or a last-node pipe end can be a far-end junction.

M. Subroutine RUPDSK(J, PRDB, PRDG, KRD)

The RUPDSK subroutine performs the boundary calculation for the rupture disk with burst pressure PRDB located at junction number JUN(J), using Eqs. 44 and 45. If KRD = 0, the disk has not burst and the junction is computed as a closed end. If KRD = 1, the disk has burst previously and the junction is computed as a constant-pressure boundary at pressure PRDG. If KRD = 0 and the calculated pressure exceeds PRDB, KRD is set equal to one and the computation is repeated as a burst-disk case; the change in KRD is detected in the main program, which prints a message giving the location of the burst disk and the time of rupture. The pipe end connected at the rupture disk can be either a first- or last-node end.

N. Subroutine PRESSO(J,PSO)

The subroutine PRESSO performs the boundary calculation at the pressure source at pressure PSO and located at junction number JUN(J), using Eqs. 46 and 47. As many as six pipes can be connected to the pressure-source junction at either node end.

O. Subroutine PTIME(T,P)

The PTIME subroutine interpolates linearly for source pressure P as a function of time T in a table of values of pressure PTM(K) at time TIME(K), where K = 1, 2, ..., NOPT. The table is input originally to the main program and is contained in COMMON. For times before the first tabulated point TIME(1), the pressure is set equal to the first tabulated value; i.e., P = PTM(1). For times greater than the last tabulated point, i.e., T > TIME(NOPT), the pressure is set equal to the last tabulated value; i.e., P = PTM(NOPT). PTIME is called from the "Initialization at TBEG" and "Initialization for Next Time Step" subsections of the main program.

V. PTA-1 INPUT REQUIREMENTS

Card No.	FORTRAN Name	Format	Description
1		16I5	
	NPIPE		Number of pipes. 1 ≤ NPIPE ≤ LMAX = 100.
	NOJUN		Number of junctions. 1 ≤ NOJUN ≤ JMAX = 100.
	NOTNK		Number of surge tanks. 0 ≤ NOTNK ≤ NTNKMX = 10.
	NORD		Number of rupture disks. 0 ≤ NORD ≤ NRDMX = 10.
	NOPUMP		Number of pumps. 0 ≤ NOPUMP ≤ NPMPMX = 10.
	KFRIC		KFRIC = 1, pipe friction is included. KFRIC = 0, pipe friction is not included.
	NOPT		Number of pressure-time points specifying pressure source. 0 ≤ NOPT ≤ NPTMX = 50.
	INCOND		INCOND = 1, initial velocities and pressures are input individually for all pipes. INCOND = 0, initial velocities and pressures are all set to UR and PR, respectively.

Card No.	FORTRAN Name	Format	Description
	NOPRIN		Frequency of printout. If NOPRIN = 1, results for every time step are printed; if NOPRIN = 2, results for every second time step are printed; etc.
	IPRIN		Detail of printout. Results are printed for every IPRINth node of each pipe specified below.
	NLPRN		NLPRN = 0, results for all pipes are printed. NLPRN = 6, e.g., results for six pipes are printed (pipe numbers specified on card 5). 0 ≤ NLPRN ≤ NPIPE.
2		5E10.5	
	DT		Time step, seconds.
	TBEG		Time at which calculation begins, seconds.
	TFIN		Time at which calculation terminates, seconds.
	UR		Initial velocity, ft/sec. If INCOND = 0, initial velocities are set to UR in all pipes.
	PR		Initial pressure, psi. If INCOND = 0, initial pressures are set to PR in all pipes.
	TEMP		Fluid temperature, °F.
3a		4I5,5E10.5	The set of cards 3a, 3b is repeated for L = 1, NPIPE.
	LPIPE(L)		Pipe number (arbitrary).
	J1(L)		Junction number at first node of pipe.
	JN(L)		Junction number at last node of pipe.
	MAT(L)		Material number of pipe: MAT = 1, Type 304 stainless steel; MAT = 2, Type 316 stainless steel; MAT = 3, linear elastic and plastic regions with elliptic transition; MAT = 4, bilinear stress-strain relation; MAT = 5, same as 4 with different material constants; MAT = 6, regid pipe wall.
	D(L)		Inner diameter of pipe, inches.
	H(L)		Wall thickness of pipe, inches.
	PLNGTH(L)		Pipe length. feet.
	RF(L)		Wall roughness, inches.
	A(L)		Flow area, in.^{2} If $A = \pi D^2/4$, set A = 0 and it will be computed from D.
3b		4E10.5	Include cards 3b if INCOND = 1; omit if INCOND = 0.
	PO1(L)		Initial pressure at first node of pipe, psi.
	PON(L)		Initial pressure at last node of pipe, psi.
	UO1(L)		Initial velocity at first node of pipe, ft/sec.
	UON(L)		Initial velocity at last node of pipe, ft/sec.
4		16I5	J = 1, NOJUN (eight junctions per card).
	JUN(J)		Junction number (arbitrary).

Card No.	FORTRAN Name	Format	Description
	JTYPE(J)		Junction type: JTYPE = 1, sudden expansion or contraction; JTYPE = 3, tee (three to six branches); JTYPE = 4, pump; JTYPE = 6, acoustic-impedance discontinuity (no area change) or dummy junction; JTYPE = 7, closed end; JTYPE = 8, surge tank (constant-pressure boundary); JTYPE = 9, far end (nonreflecting); JTYPE = 10, rupture disk; JTYPE = 15, pressure-pulse source.
5		16I5	K = 1, NLPRN; omit if NLPRN = 0.
	LPRIN(K)		Pipe numbers for which results are printed.
6		I10,E10.5	K = 1, NOTNK; omit if NOTNK = 0.
	JTANK(K)		Junction number to which surge tank or constant-pressure boundary is connected.
	PTANK(K)		Gas pressure of surge tank or constant-pressure boundary, psi.
7		I10,2E10.5	K = 1, NORD; omit if NORD = 0.
	JRD(K)		Junction number to which rupture disk is connected.
	PRDB(K)		Burst pressure of rupture disk, psi.
	PRDG(K)		Gas pressure behind disk, psi.
8		2I5,E10.5	K = 1, NOPUMP; omit if NOPUMP = 0.
	JPUMP(K)		Junction number to which pump is connected.
	LPUMP(K)		Pump-discharge pipe number.
	HEAD(K)		Pump head, psi.
9		8E10.5	K = 1, NOPT; omit if NOPT = 0.
	TIME(K)		Times for which source-pulse data are input, seconds.
10		8E10.5	K = 1, NOPT; omit if NOPT = 0.
	PTM(K)		Source-pulse pressure at time TIME(K), psi.

The values of LMAX, JMAX, NTNKMX, NRDMX, NPMPMX, and NPTMX, which are given in prescribing limits on input data on card 1, are those contained in the listed version of PTA-1. (See Sec. VII for information on increasing these limits to accommodate a larger pipe network or decreasing them to conserve core storage.)

If INCOND = 0, the initial velocities and pressures throughout the system are all set to UR and PR, respectively, where UR and PR are input on card 2. Cards 3b are omitted in this case. If INCOND = 1, initial velocities and pressures at the first and last nodes of each pipe are input on cards 3b. Initial velocities and pressures at intermediate nodes are then computed in the program, using linear interpolation between the end-node values.

Pipe and junction numbers may be assigned arbitrarily, and either end of a pipe may be designated as the first-node end.

The diameter D and flow area A of each pipe are input independently on cards 3a to accommodate the modeling of a component where $A \neq \pi D^2/4$. The diameter is used in computing deformation response through the ratio H/D and pipe-friction losses, and the area is used in computing junction conditions at tees, pumps, and sudden expansions or contractions. An input of A = 0 will result in the area being computed from $A = \pi D^2/4$.

If KFRIC = 0 on card 1, i.e., pipe friction is not included in the problem, it is not necessary to input values of pipe roughness RF(L) on cards 3a.

VI. DIAGNOSTIC MESSAGES

The diagnostic messages printed by PTA-1 are listed in this section along with some explanations of their implications.

"ERROR IN NUMBER OF PIPES"

The input value on card 1 of the number of pipes NPIPE in the system is less than one or greater than LMAX, where LMAX = 100 in the listed version of PTA-1.

"ERROR IN NUMBER OF JUNCTIONS"

The input value on card 1 of the number of junctions NOJUN in the system is less than one or greater than JMAX, where JMAX = 100 in the listed version of PTA-1.

"ERROR IN NUMBER OF TANKS"

The input value on card 1 of the number of constant-pressure surge tanks NOTNK in the system is negative or greater than NTNKMX, where NTNKMX = 10 in the listed version of PTA-1.

"NOTNK - 0 AND JUNCTION xxx IS A SURGE TANK"

The input value on card 1 of NOTNK = 0 indicates that there are no surge tanks in the system, but junction number xxx is identified as type 8 (surge-tank junction) on card 4.

"NO TANK DATA INPUT FOR JUNCTION xxx"

Junction number xxx is identified as type 8 (surge-tank junction) on card 4, but no surge-tank data have been input on card 6 for this junction.

"ERROR IN NUMBER OF RUPTURE DISKS"

The input value on card 1 of the number of rupture disks NORD in the system is negative or greater than NRDMX, where NRDMX = 10 in the listed version of PTA-1.

"NORD = 0 AND JUNCTION xxx IS A RUPTURE DISK"

The input value of NORD = 0 on card 1 indicates that there are no rupture disks in the system, but junction number xxx is type 10 (rupture-disk junction) on card 4.

"NO RUPTURE DISK DATA INPUT FOR JUNCTION xxx"

Junction number xxx is identified as type 10 (rupture-disk junction) on card 4, but no rupture-disk data have been input on card 7 for this junction.

"ERROR IN NUMBER OF PUMPS"

The input value on card 1 of the number of pumps NOPUMP in the system is negative or greater than NPMPMX, where NPMPMX = 10 in the listed version of PTA-1.

"PUMP DISCHARGE PIPE NUMBER IS INCORRECT JPUMP = xxx, LPUMP = yyy"

On card 8, a pump is located at junction number xxx having discharge pipe number yyy. This error message results if either (1) no pipe number yyy occurs in the system, as evidenced by the values of LPIPE on cards 3a, or (2) pipe number yyy is not connected to junction number xxx, which is discovered by comparing junction number xxx with the junction numbers J1(L) and JN(L) associated with pipe number LPIPE(L) = yyy on card 3a.

"NOPUMP = 0 AND JUNCTION xxx IS A PUMP"

The input value on card 1 of NOPUMP = 0 indicates that there are no pumps in the system, but junction number xxx is type 4 (pump junction) on card 4.

"NO PUMP DATA INPUT FOR JUNCTION xxx"

Junction number xxx is identified as type 4 (pump junction) on card 4, but no pump data on card 8 have been input for this junction.

"ERROR IN NUMBER OF PRESSURE SOURCE DATA POINTS"

The input value on card 1 of the number of pressure-source data points NOPT is negative or greater than NPTMX, where NPTMX = 50 in the listed version of PTA-1.

"NO INPUT PRESSURE PULSE (NOPT = 0) AT SOURCE JUNCTION xxx"

The input value on card 1 of NOPT = 0 indicates that there is no pressure-pulse source in the system, but junction number xxx is type 15 (pressure-pulse source) on card 4.

"INCORRECT PIPE ARRANGEMENT AT JUNCTION xxx, JTYPE = yyy"

This message is printed when the number of pipes connected at junction number xxx is inappropriate for the associated junction type yyy given on card 4. In particular, it occurs when more or less than one pipe is connected to a single pipe boundary; more or less than two pipes are connected at a two-pipe junction; or more than NBRMX pipes are connected at a multipipe junction, where NBRMX = 6 in the listed version of PTA-1.

"INVALID JUNCTION TYPE, JUN = xxx, JTYPE = yyy"

The junction type yyy associated with junction number xxx on card 4 does not correspond to a permissible junction type.

"PIPES CONNECTED AT JUNCTION xxx HAVE DIFFERENT AREAS, JTYPE = 6"

According to card 4, junction number xxx is junction type 6 (acoustic-impedance discontinuity or dummy junction), but the pipes connected there have different areas. The junction probably should be type 1 (sudden expansion or contraction junction).

"NNODE.GT.IMAX FOR PIPE NO. xxx--RESET DT = yyy TO DT = zzz"

For the input time step of DT = yyy on card 2, the computed number of nodes NNODE for pipe number xxx exceeds IMAX, where IMAX = 100 in the listed version of PTA-1. The time step has been increased to DT = zzz, and problem execution is continued.

"CUMX = xxx IS GREATER THAN FACT2"

CUMX > F_2 (see Sec. III.C), where CUMX is the maximum value of $(c + |u|)/c$ occurring in the system. Since numerical instability is likely to result, the end-of-problem printouts are written and execution is terminated.

"*****DT IS DECREASED TO xxx*****"

$F_1 \leq$ CUMX $\leq F_2$ (see Sec. III.C), where CUMX is the maximum value of $(c + |u|)/c$ occurring in the system. Since the Courant-Friedrichs-Lewy criterion may be violated if the current time step is used for subsequent computations, DT is decreased to xxx and execution is continued.

"NUMBER OF PLASTIC PIPES EXCEEDS NPLMX"

The number of pipes in the system that have experienced some plastic deformation exceeds NPLMX, where NPLMX = 10 in the listed version of PTA-1. Since the storage allocations for plasticity-related quantities are consistent with NPLMX, end-of-problem printouts are written and execution is terminated.

"NO CONVERGENCE IN PLASTIC ITERATION FOR PIPE NO. xxx, I = yyy, TIME = zzz SEC"

The iterative procedure for calculating pressure and velocity in a plastically deforming pipe does not converge at node number yyy of pipe number LPIPE = xxx at time zzz. End-of-problem printouts are written, and execution is terminated.

"NO CONVERGENCE IN PLASTIC ITERATION AT JUNCTION xxx, TIME = yyy SEC"

The iterative procedure for calculating pressure and velocity in a plastically deforming pipe does not converge at junction number xxx at time yyy. End-of-problem printouts are written, and execution is terminated.

"INCORRECT MATERIAL NUMBER, MAT = xxx"

This message is written by the materials-property subroutine MTPRP and indicates that material type number xxx is outside the permissible range 1-6.

"NO CONVERGENCE IN INTRP ITERATION LPIPE = xxx, I = yyy, K = z"

This message is written by the interpolation subroutine INTRP and indicates that the interpolation procedure at node number yyy of plastically deforming pipe number xxx does not converge. If K = 1, the interpolation is along the C^+ characteristic (point A of Figs. 1 and 2); if K = 2, the interpolation is along the C^- characteristic (point B of Figs. 1 and 2).

"NO CONVERGENCE IN FRICTION TERM FOR LPIPE = xxx, VELOCITY = yyy"

This message is written by the friction-term subroutine FRCTRM and indicates that there is no convergence in the iterative computation of the pipe friction factor for pipe number xxx at velocity yyy.

"AREACH SUBROUTINE--SQUARE ROOT HAS NEGATIVE ARGUMENT AT JUNCTION xxx"

This message is printed by the area-change junction subroutine and indicates that the square-root term in Eqs. 31 has a negative argument for junction number xxx.

VII. ALTERATIONS TO PROGRAM STORAGE

The storage allotments in the listed version of PTA-1 are consistent with limitations given in Table II and determine the complexity of the piping network which can be analyzed. The values of LMAX, JMAX, etc., given in Table II, are prescribed in DATA statements in the main program; and arrays in COMMON and DIMENSION statements in the main program and subroutines are sized in accordance with them.

TABLE II. Storage Limits

Name	Description	Value
LMAX	Maximum number of pipes	100
JMAX	Maximum number of junctions	100
IMAX	Maximum number of nodes per pipe	100
NBRMX	Maximum number of pipes connected at a multi-branch junction	6
NPLMX	Maximum number of plastically deforming pipes	10
NPTMX	Maximum number of pressure-time input points	50
NTNKMX	Maximum number of constant-pressure surge tanks	10
NRDMX	Maximum number of rupture disks	10
NPMPMX	Maximum number of pumps	10

Array sizes can be altered either to increase the complexity and size of the piping network being analyzed or to reduce core-storage requirements. The changes in COMMON and DIMENSION storage allocations compatible with changes in the limits LMAX, JMAX, etc., are discussed below.

A. Array Sizes in COMMON Statements

The sharings of numbered common blocks between the main program and subroutine are listed in Table I. The COMMON statements, with limit names substituted for numerical values in the storage allocations, are:

```
COMMON/R1/ A(LMAX),C(LMAX),DZ(LMAX),D(LMAX)
           P(LMAX,IMAX,2),U(LMAX,IMAX,2),LJ1(JMAX,NBRMX),
           LJN(JMAX,NBRMX),ML1(JMAX),MLN(JMAX),
           LPIPE(LMAX),JUN(JMAX),NNODE(LMAX),DENS,DT

COMMON/R2/ RF(LMAX),VISC,KFRIC

COMMON/R3/ TEMP,CO

COMMON/R4/ PMX(NPLMX,IMAX),KPLAS(LMAX),HOD(LMAX),
           MAT(LMAX),PY(LMAX),CPX(NPLMX),MPX(NPLMX)

COMMON/R5/ NOPT,TIME(NPTMX),PTM(NPTMX)
```

B. Array Sizes in DIMENSION Statements

The DIMENSION statements in the main program and subroutines are as follows, with limit names substituted for numerical values in the storage allocations.

1. Main Program

DIMENSION PLNGTH(LMAX),H(LMAX)

DIMENSION LPRIN(LMAX),PMAX(LMAX),TMAX(LMAX)

DIMENSION PO1(LMAX),PON(LMAX),UO1(LMAX),UON(LMAX)

DIMENSION JTYPE(JMAX),J1(LMAX),JN(LMAX)

DIMENSION LPLP(NPLMX),PPIJ(NPLMX),UPIJ(NPLMX)

DIMENSION PTANK(NTNKMX),JTANK(NTNKMX)

DIMENSION PRDB(NRDMX),PRDG(NRDMX),KRD(NRDMX),
JRD(NRDMX),TRD(NRDMX)

DIMENSION JPUMP(NPMPMX),LPUMP(NPMPMX),
HEAD(NPMPMX),LLPMP(NPMPMX)

2. Subroutine MTPRP

DIMENSION E(6),SY(6),KPR(6),EPSY(6) (Compatible with number of material representations)

3. Subroutine FLPRP

None.

4. Subroutine INTRP

None.

5. Subroutine FRCTRM

None.

6. Subroutine AREACH

None.

7. Subroutine TEE

DIMENSION UA(NBRMX),UB(NBRMX),PA(NBRMX),PB(NBRMX),
CA(NBRMX),CB(NBRMX),GA(NBRMX),GB(NBRMX)

8. Subroutine PUMP

DIMENSION UA(NBRMX),UB(NBRMX),PA(NBRMX),PB(NBRMX),
CA(NBRMX),CB(NBRMX),GA(NBRMX),GB(NBRMX)

9. Subroutine IMPED

None.

10. Subroutine CLOSED

None.

11. Subroutine CONSTP

None.

12. Subroutine FAREND

None.

13. Subroutine RUPDSK

None.

14. Subroutine PRESSO

None.

15. Subroutine PTIME

None.

APPENDIX A

Sample Problem

The following sample problem is included to illustrate input requirements for PTA-1 and to demonstrate the effect of plastic deformation on pulse propagation.

Consider the simple piping network shown in Fig. 4. Encircled numbers are the pipe identification numbers; the notation 18", 50' means that the pipe is 18 in. in diameter and 50 ft long; and the notation (10,3) indicates that junction number 10 is type 3, i.e., a tee junction. Junctions 1 and 2 could have been assigned the same number since they are both connected to the source. The 18-in. pipes have a 0.438-in. wall thickness, and the wall of the 24-in. pipe is 0.562 in. thick. All pipes are made of Type 304 stainless steel (MAT = 1) and have a wall roughness of 0.005 in. The sodium coolant is at 700°F and is initially at rest under 100-psi pressure. The rupture disk at junction 11 has a burst pressure of 300 psi, with a 15-psi reservoir behind it.

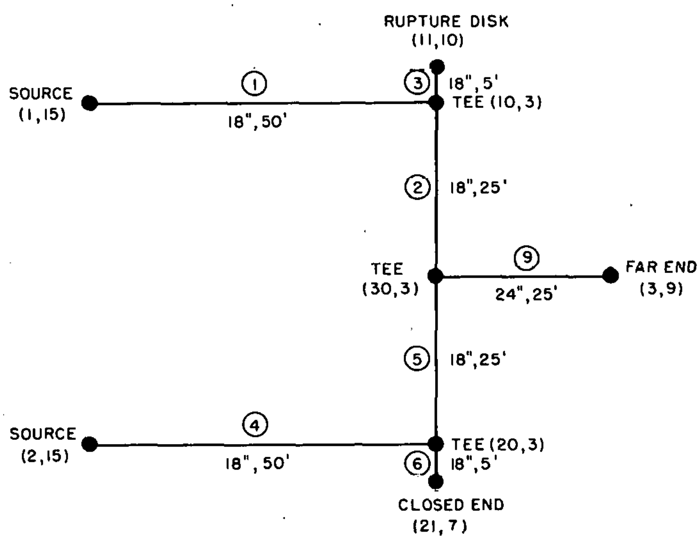


Fig. 4. Piping Network for Sample Problem

Figure 5 shows the input data for the sample problem: A triangular pressure pulse having a rise to 1100 psi in 10 msec, followed by a decay to 100 psi in an additional 40 msec, is specified. The time step is 0.5 msec, and the transient is initiated at t = 0 and followed to t = 75 msec. Results for every other axial node of each pipe are printed for every fifth time step. Cross-sectional areas of the pipes are not input since they can be calculated from the diameters. Cards 3b are omitted because initial velocities and pressures are the same for all pipes; Card 5 is omitted because results are printed for all pipes; and Cards 6 and 8 are omitted because there are no surge tanks or pumps in the system.

The computation indicates that the rupture disk burst at 12 msec and pipes 1, 4, 5, and 6 deformed plastically during the transient. Figures 6 and 7 give the pressure histories at the tee at junction 20 and the nonreflecting end at junction 3, respectively. The source pulse is also shown on Fig. 6.

To demonstrate the effect of plastic deformation on transient response, the same problem was run with all pipes made of material 5, which has the same elastic modulus as Type 304 stainless steel but does not deform plastically at system pressures. The dashed curves on the figures show the resulting elastic response.

COMPUTER INPUT DATA FORM

COST CODE ____________

PROGRAM PTA1   PROBLEM SAMPLE   ORIGINATOR   DATE   PAGE   OF

(CARD NO.)	Data
(1)	7 8 0 1 0 1 3 0 5 2 0
(2)	0.0005 0.0 0.075 0.0 100.0 700.0
(3a)	1 1 10 1 18.0 0.438 50.0 0.005
	4 2 20 1 18.0 0.438 50.0 0.005
	2 10 30 1 18.0 0.438 25.0 0.005
	5 30 20 1 18.0 0.438 25.0 0.005
	9 30 3 1 24.0 0.562 25.0 0.005
	3 10 11 1 18.0 0.438 5.0 0.005
	6 20 21 1 18.0 0.438 5.0 0.005
(4)	1 15 10 3 30 3 20 3 2 15 3 9 11 10 21 7
(7)	11 300.0 15.0
(9)	0.0 0.010 0.050
(10)	100.0 1100.0 100.0

AMD-9 (6-65)

Fig. 5. Input Data for Sample Problem

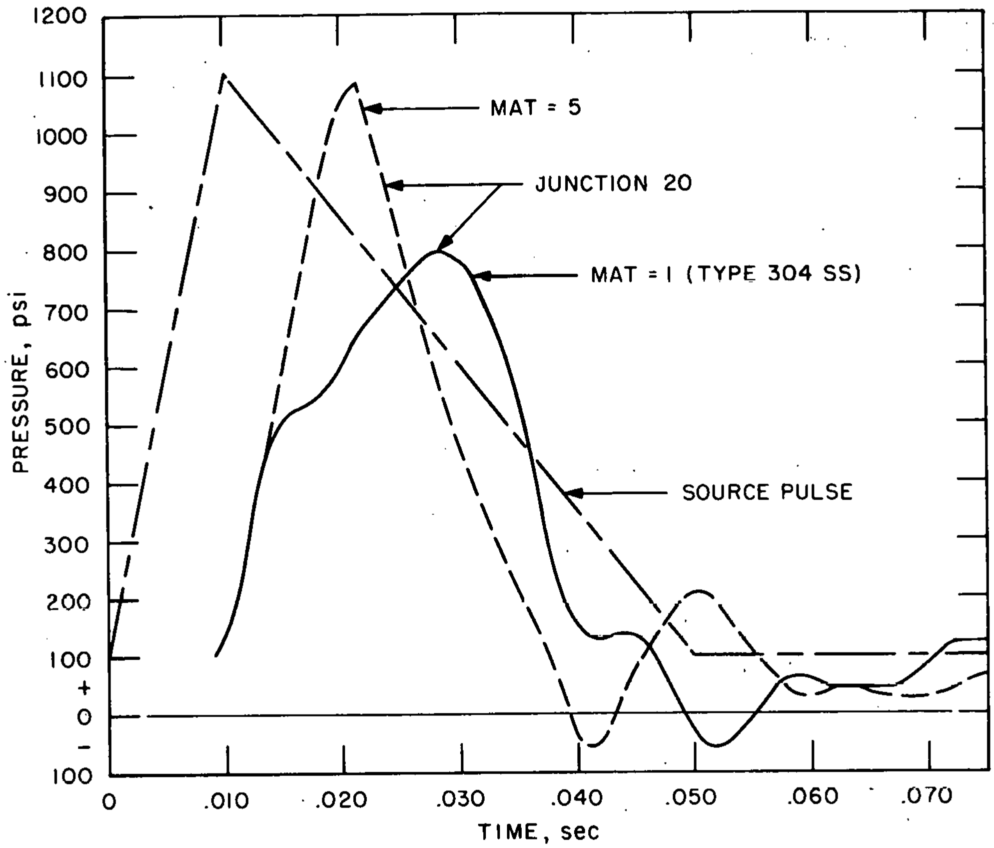


Fig. 6. Pressure History at Junction 20 and Source Pulse at Junctions 1 and 2

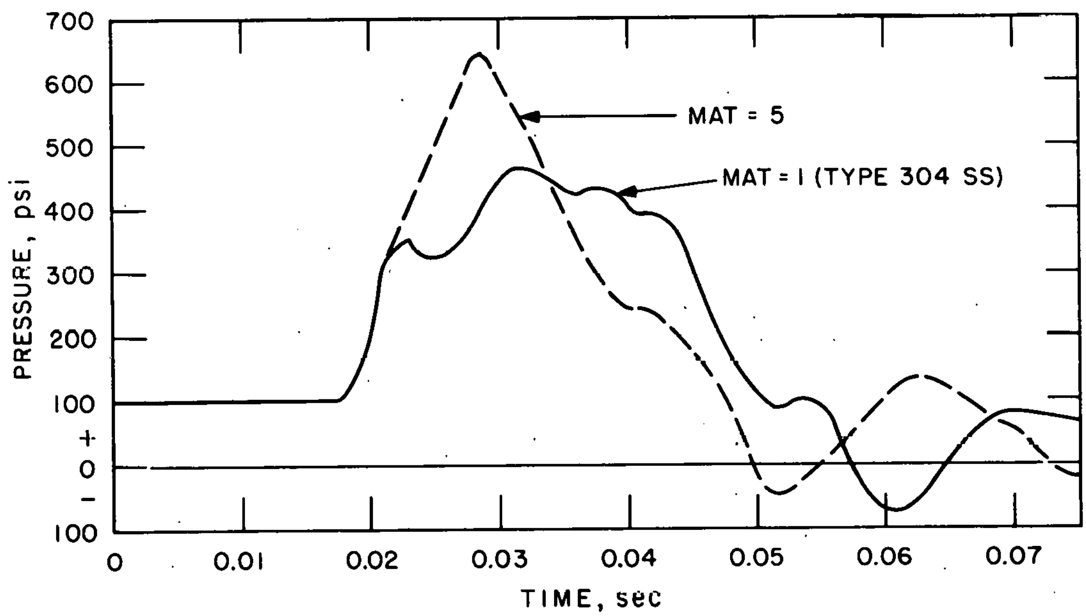


Fig. 7. Pressure History at Junction 3

The departure from linearity of the stress-strain curve for Type 304 stainless steel occurs at pressures of about 500 psi for pipes of the given dimensions. The leading edge of the pulse up to this pressure is consequently unaffected by plastic deformation. Since the burst pressure of the rupture disk is less than the yield pressure, the disk bursts at the same time for both problems. The effect of plastic deformation is to lower the peak pressure at junction 20 by about 300 psi and to lower the peak pressure at junction 3 by about 200 psi. The peaks are also delayed.

Note that negative pressures are computed at later times, indicating that cavitation would occur in the system. Subsequent versions of the PTA code will treat the effects of cavitation on transient propagation.

APPENDIX B

Listing of PTA-1

```
C
C
C     **************************************************************************
C     *                          PTA-1 PROGRAM                                 *
C     *                  PRESSURE TRANSIENT ANALYSIS                           *
C     *    THE METHOD OF CHARACTERISTICS IS USED FOR PRESSURE TRANSIENT        *
C     *    ANALYSIS IN HYDRAULIC NETWORKS FOR ARBITRARY PRESSURE SOURCE.       *
C     *    THE PROGRAM CONSIDERS COMPLETE ONE-DIMENSIONAL EQUATIONS            *
C     *    INCLUDING NONLINEAR CONVECTIVE TERMS AS IN THE PROGRAM              *
C     *    NATRANSIENT (Y.W. SHIN, ANL-8049).  THE EFFECT OF PLASTIC          *
C     *    PIPE DEFORMATION ON PULSE PROPAGATION IN THE FLUID IS               *
C     *    ACCOUNTED FOR AS IN THE PROGRAM PLWV (C.K. YOUNGDAHL AND            *
C     *    C.A. KOT, ANL-75-5).                                                *
C     *    C.K. YOUNGDAHL - OCTOBER 1976                                       *
C     **************************************************************************
C
C
C****************************** MAIN PROGRAM **********************************
C
C
C     ***************** PARTIAL LISTING OF INPUT DATA ************************
C     *   NPIPE    -  NUMBER OF PIPES                                          *
C     *   NOJUN    -  NUMBER OF JUNCTIONS                                      *
C     *   NOTNK    -  NUMBER OF CONSTANT PRESSURE TANKS                        *
C     *   NORD     -  NUMBER OF RUPTURE DISKS                                  *
C     *   NOPUMP   -  NUMBER OF PUMPS                                          *
C     *   KFRIC    -  IF KFRIC = 0, INVISCID ANALYSIS                          *
C     *               IF KFRIC = 1, PIPE FRICTION IS INCLUDED                  *
C     *   NOPT     -  NUMBER OF DATA POINTS FOR PRESCRIBED PRESSURE SOURCE     *
C     *   INCOND   -  IF INCOND = 1, INITIAL CONDITIONS ARE INPUT INDIVIDUALLY *
C     *   NOPRIN   -  FREQUENCY OF PRINTOUT                                    *
C     *   IPRIN    -  PARAMETER SPECIFYING DETAIL OF PRINTOUT FOR EACH PIPE    *
C     *   NLPRN    -  NUMBER OF PIPES FOR WHICH RESULTS ARE PRINTED            *
C     *               IF NLPRN = 0, RESULTS ARE PRINTED FOR ALL PIPES          *
C     *   DT       -  TIME STEP IN SECONDS                                     *
C     *   TBEG     -  TIME AT WHICH CALCULATION BEGINS, SECONDS                *
C     *   TFIN     -  FINAL TIME AT WHICH CALCULATION TERMINATES, SECONDS      *
C     *   UR       -  INITIAL VELOCITY IN FT/SEC                               *
C     *   PR       -  INITIAL PRESSURE IN PSI                                  *
C     *   TEMP     -  SYSTEM TEMPERATURE IN DEGREES FAHRENHEIT                 *
C     *   JTANK    -  JUNCTION TO WHICH TANK IS CONNECTED                      *
C     *   PTANK    -  CONSTANT PRESSURE AT TANK, PSI                           *
C     *   JRD      -  JUNCTION TO WHICH RUPTURE DISK IS CONNECTED              *
C     *   PRDB     -  BURST PRESSURE OF RUPTURE DISK, PSI                      *
C     *   PRDG     -  GAS PRESSURE BEHIND DISK, PSI                            *
C     *   JPUMP    -  PUMP JUNCTION NUMBER                                     *
C     *   LPUMP    -  PUMP DISCHARGE PIPE NUMBER                               *
C     *   HEAD     -  PUMP HEAD, PSI                                           *
C     **************************************************************************
C
C     ***************** DEFINITIONS OF JUNCTION TYPE NUMBERS *****************
C     *    1       -  SUDDEN EXPANSION OR CONTRACTION                          *
C     *    3       -  TEE JUNCTION (3 TO 6 BRANCHES), NO LOSSES                *
```

```
C     *     4          -  PUMP JUNCTION                                                      *
C     *     6          -  ACOUSTIC-IMPEDANCE DISCONTINUITY (NO AREA CHANGE)                  *
C     *     6          -  DUMMY JUNCTION                                                     *
C     *     7          -  CLOSED END                                                         *
C     *     8          -  CONSTANT PRESSURE BOUNDARY  (SIMPLE SURGE TANK)                    *
C     *     9          -  FAR END  (NONREFLECTING)                                           *
C     *    10          -  RUPTURE DISC  (INSTANTANEOUS-CONSTANT PRESSURE)                    *
C     *    15          -  PRESSURE SOURCE JUNCTION                                           *
C     ********************************************************************************
C
      COMMON  /R1/  A(100),C(100),DZ(100),D(100),P(100,100,2),
     1              U(100,100,2),LJ1(100,6),LJN(100,6),ML1(100),MLN(100),
     2              LPIPE(100),JUN(100),NNODE(100),DENS,DT
      COMMON  /R2/  RF(100),VISC,KFRIC
      COMMON  /R3/  TEMP,CO
      COMMON  /R4/  PMX(10,100),KPLAS(100),HOD(100),MAT(100),PY(100),
     1              CPX(10),MPX(10)
      COMMON  /R5/  NOPT,TIME(50),PTM(50)
C
      DIMENSION  PLNGTH(100),H(100)
      DIMENSION  LPRIN(100),PMAX(100),TMAX(100)
      DIMENSION  PO1(100),PON(100),UO1(100),UON(100)
      DIMENSION  JTYPE(100),J1(100),JN(100)
      DIMENSION  LPLP(10),PPIJ(10),UPIJ(10)
      DIMENSION  PTANK(10),JTANK(10)
      DIMENSION  PRDB(10),PRDG(10),KRD(10),JRD(10),TRD(10)
      DIMENSION  JPUMP(10),LPUMP(10),HEAD(10),LLPMP(10)
C
      DATA  JMAX/100/,LMAX/100/,IMAX/100/,NPTMX/50/
      DATA  NTNKMX/10/,NRDMX/10/,NPLMX/10/,NBRMX/6/,NPMPMX/10/
      DATA  KIN,KOUT/5,6/,K1,K2/1,2/,EPSI/0.0001/
      DATA  FACT1,FACT2/1.03,1.10/,AFAC,XFAC/144.0,12.0/
C
C
C***********************************  INPUT  ************************************
C
C**** READ PROBLEM AND CONTROL CONSTANTS
      READ(KIN,900) NPIPE,NOJUN,NOTNK,NORD,NOPUMP,KFRIC,NOPT,
     X     INCOND,NOPRIN,IPRIN,NLPRN
      IF (NPIPE.GT.LMAX.OR.NPIPE.LT.1)  GO TO 800
      IF (NOJUN.GT.JMAX.OR.NOJUN.LT.1)  GO TO 805
C**** READ PROBLEM PARAMETERS
      READ(KIN,901)  DT,TBEG,TFIN,UR,PR,TEMP
C**** READ PIPE DESCRIPTIONS
      DO 5  L=1,NPIPE
      READ(KIN,905)  LPIPE(L),J1(L),JN(L),MAT(L),D(L),H(L),PLNGTH(L),
     X     RF(L),A(L)
      IF(INCOND.EQ.1) READ(KIN,901) PO1(L),PON(L),UO1(L),UON(L)
    5 CONTINUE
C**** READ JUNCTION DESCRIPTIONS
      READ(KIN,900)(JUN(J),JTYPE(J),J=1,NOJUN)
C**** READ PIPE NUMBERS FOR WHICH RESULTS ARE PRINTED
      IF (NLPRN.GT.0) READ(KIN,900) (LPRIN(K),K=1,NLPRN)
C**** READ CONDITIONS FOR CONSTANT PRESSURE TANKS
      IF(NOTNK.GT.NTNKMX.OR.NOTNK.LT.0) GO TO 810
```

```
      IF(NOTNK.GT.0) READ(KIN,910) (JTANK(K),PTANK(K),K=1,NOTNK)
C**** READ CONDITIONS FOR RUPTURE DISKS
      IF(NORD.GT.NRDMX.OR.NORD.LT.0) GO TO 815
      IF(NORD.EQ.0) GO TO 15
      DO 10 K=1,NORD
      READ(KIN,911) JRD(K),PRDB(K),PRDG(K)
   10 KRD(K)=0
C**** READ DATA FOR PUMPS
   15 IF(NOPUMP.GT.NPMPMX .OR. NOPUMP.LT.0) GO TO 820
      IF(NOPUMP.EQ.0) GO TO 25
      DO 19  K=1,NOPUMP
      READ(KIN,912)  JPUMP(K),LPUMP(K),HEAD(K)
      DO 16  L=1,NPIPE
      IF(LPIPE(L).EQ.LPUMP(K))  GO TO 17
   16 CONTINUE
      GO TO 822
   17 LLPMP(K)=L
      IF(J1(L).NE.JPUMP(K) .AND. JN(L).NE.JPUMP(K))  GO TO 822
   19 CONTINUE
C**** READ PRESSURE SOURCE DATA
   25 IF(NOPT.GT.NPTMX.OR.NOPT.LT.0)  GO TO 828
      IF(NOPT.EQ.0) GO TO 26
      READ(KIN,901) (TIME(K),K=1,NOPT)
      READ(KIN,901) (PTM(K),K=1,NOPT)
   26 CONTINUE
C
C
C************************  COMPUTE PROBLEM PARAMETERS  ************************
C
C**** CALCULATE SPECIFIC WEIGHT, DENSITY, VISCOSITY, SOUND SPEED
      CALL FLPRP(SPWT,DENS,VISC,CO)
C**** COMPUTE AREAS NOT GIVEN AS INPUT
      DO 100  L=1,NPIPE
      IF(A(L) .GT. 0.0001) GO TO 100
      A(L)=3.14159265*D(L)**2/4.0
  100 CONTINUE
C**** WAVE SPEED FOR ELASTIC PIPE DEFORMATION AND PIPE YIELD PRESSURE
      DO 105  L=1,NPIPE
      HOD(L)=H(L)/D(L)
      CALL ELPRP(MAT(L),HOD(L),PY(L),C(L))
  105 CONTINUE
C**** DETERMINE DZ(L) AND NNODE(L) IN EACH PIPE
  110 DO 114 L=1,NPIPE
      ZM1=FACT1*C(L)*DT
      ZM2=FACT2*C(L)*DT
      NIM=PLNGTH(L)/ZM1+0.5
      IF(NIM .EQ. 0)  NIM=1
      NNODE(L)=NIM+1
      IF(NNODE(L).LE.IMAX) GO TO 112
C**** RECALCULATE DT TO SATISFY NNODE(L) .LE. IMAX
      DTT=2.*DT
      GO TO 870
  111 DT=DTT
      GO TO 110
C**** CALCULATE DZ(L)
```

```
  112 DZ(L)=PLNGTH(L)/NIM
      IF (DZ(L) .LT. ZM1)  DZ(L)=ZM1
      IF (DZ(L) .GT. ZM2)  DZ(L)=ZM2
  114 PLNGTH(L)=NIM*DZ(L)
C
C
C********  PRINT INPUT, PROBLEM PARAMETERS, AND NETWORK ARRANGEMENT  ***********
C
      WRITE(KOUT,930) NPIPE,NOJUN,NOTNK,NORD,NOPUMP,KFRIC,NOPT,
     X    INCOND,NOPRIN,IPRIN,NLPRN
      WRITE(KOUT,932)  DT,TBEG,TFIN,UR,PR,TEMP,SPWT,VISC,CO
      IF(NOTNK.EQ.0) GO TO 150
      WRITE(KOUT,934)
      WRITE(KOUT,935)  (JTANK(K),PTANK(K),K=1,NOTNK)
  150 IF(NORD.EQ.0) GO TO 155
      WRITE(KOUT,937)
      WRITE(KOUT,938)  (JRD(K),PRDB(K),PRDG(K),K=1,NORD)
  155 IF(NOPUMP.EQ.0)  GO TO 160
      WRITE(KOUT,940)
      WRITE(KOUT,941) (JPUMP(K),LPUMP(K),HEAD(K),K=1,NOPUMP)
  160 WRITE(KOUT,945)
      DO 166  L=1,NPIPE
      DO 162  J=1,NOJUN
      IF(JUN(J).EQ.J1(L)) JT1=JTYPE(J)
      IF(JUN(J).EQ.JN(L)) JT2=JTYPE(J)
  162 CONTINUE
      IF(INCOND.EQ.1) GO TO 164
      WRITE(KOUT,946)  LPIPE(L),J1(L),JT1,JN(L),JT2,PR,PR,UR,UR
      GO TO 166
  164 WRITE(KOUT,946)  LPIPE(L),J1(L),JT1,JN(L),JT2,
     X    PO1(L),PON(L),UO1(L),UON(L)
  166 CONTINUE
      WRITE(KOUT,950)
      WRITE(KOUT,951)(LPIPE(L),D(L),H(L),A(L),PLNGTH(L),DZ(L),
     X    NNODE(L),L=1,NPIPE)
      WRITE(KOUT,953)
      DO 170 L=1,NPIPE
      PYX=PY(L)/AFAC
  170 WRITE(KOUT,954) LPIPE(L),MAT(L),PYX,C(L),RF(L)
      CALL WRMAT(MAT(1))
      DO 174  L=2,NPIPE
      LM=L-1
      DO 172  LL=1,LM
      IF(MAT(L).EQ.MAT(LL)) GO TO 174
  172 CONTINUE
      CALL WRMAT(MAT(L))
  174 CONTINUE
      IF(NOPT.EQ.0) GO TO 180
      WRITE(KOUT,956)
      WRITE(KOUT,957)  (TIME(K),PTM(K),K=1,NOPT)
  180 CONTINUE
C
C
C***********  DETERMINATION OF PIPE CONNECTIONS AT EACH JUNCTION  *************
C
```

```
      DO 240  J=1,NOJUN
      JT=JTYPE(J)
      M1=0
      MN=0
      DO 202  K=1,6
      LJ1(J,K)=0
  202 LJN(J,K)=0
      DO 206  L=1,NPIPE
      IF(JUN(J).NE.J1(L)) GO TO 204
      M1=M1+1
      LJ1(J,M1)=L
  204 IF(JUN(J).NE.JN(L)) GO TO 206
      MN=MN+1
      LJN(J,MN)=L
  206 CONTINUE
      ML1(J)=M1
      MLN(J)=MN
C**** TEST CORRECTNESS OF PIPE ARRANGEMENTS
      IF(M1+MN.LE.0) GO TO 830
      IF(JT.EQ.1) GO TO 225
      IF(JT.EQ.3) GO TO 230
      IF(JT.EQ.4) GO TO 234
      IF(JT.EQ.6) GO TO 226
      IF(JT.EQ.7) GO TO 210
      IF(JT.EQ.8) GO TO 215
      IF(JT.EQ.9) GO TO 210
      IF(JT.EQ.10) GO TO 220
      IF(JT.EQ.15) GO TO 232
      GO TO 835
C**** SINGLE PIPE ENDS
  210 IF(M1+MN .NE. 1) GO TO 830
      GO TO 240
C**** CONSTANT PRESSURE JUNCTION (SURGE TANK)
  215 IF(M1+MN .NE. 1) GO TO 830
      IF(NOTNK.EQ.0) GO TO 812
      GO TO 240
C**** RUPTURE DISK
  220 IF(M1+MN .NE. 1) GO TO 830
      IF(NORD.EQ.0) GO TO 817
      GO TO 240
C**** TWO PIPE JUNCTIONS
  225 IF(M1+MN .NE. 2) GO TO 830
      GO TO 240
C**** DUMMY JUNCTION OR ACOUSTIC IMPEDANCE DISCONTINUITY
  226 IF(M1+MN .NE. 2) GO TO 830
      IF(M1.GE.1) L1=LJ1(J,1)
      IF(MN.GE.1) LN=LJN(J,1)
      IF(M1.EQ.2) LN=LJ1(J,2)
      IF(MN.EQ.2) L1=LJN(J,2)
      IF(ABS(A(L1)/A(LN)-1.0).GT.EPSI) GO TO 837
      GO TO 240
C**** TEE JUNCTION
  230 IF(M1.GT.NBRMX.OR.MN.GT.NBRMX)  GO TO 830
      GO TO 240
C**** SOURCE JUNCTION
```

```
  232 IF(M1.GT.NBRMX.OR.MN.GT.NBRMX)  GO TO 830
      IF(NOPT.EQ.0) GO TO 840
      GO TO 240
C**** PUMP JUNCTION
  234 IF(M1.GT.NBRMX.OR.MN.GT.NBRMX)  GO TO 830
      IF(NOPUMP.EQ.0) GO TO 824
      GO TO 240
  240 CONTINUE
C
C
C******************  CONVERSION OF UNITS - INCHES TO FEET  *********************
C
      PR=PR*AFAC
      DO 250 L=1,NPIPE
      A(L)=A(L)/AFAC
      D(L)=D(L)/XFAC
      RF(L)=RF(L)/XFAC
      IF(INCOND.NE.1) GO TO 250
      PO1(L)=PO1(L)*AFAC
      PON(L)=PON(L)*AFAC
  250 CONTINUE
      IF(NOTNK.EQ.0) GO TO 260
      DO 255  K=1,NOTNK
  255 PTANK(K)=PTANK(K)*AFAC
  260 IF(NORD.EQ.0) GO TO 265
      DO 262  K=1,NORD
      PRDB(K)=PRDB(K)*AFAC
  262 PRDG(K)=PRDG(K)*AFAC
  265 IF(NOPUMP.EQ.0) GO TO 270
      DO 266  K=1,NOPUMP
  266 HEAD(K)=HEAD(K)*AFAC
  270 IF(NOPT.EQ.0) GO TO 272
      DO 271  K=1,NOPT
  271 PTM(K)=PTM(K)*AFAC
  272 CONTINUE
C
C
C***********************  INITIALIZATIONS AT TBEG  ****************************
C
      NSTEP=0
      TT=TBEG
      DO 302  L=1,NPIPE
      TMAX(L)=TBEG
  302 PMAX(L)=0.0
C**** INITIAL VALUES OF UO, PO, U AND P
      DO 315  L=1,NPIPE
      NI=NNODE(L)
      IF(INCOND.EQ.1) GO TO 307
      DO 305  I=1,NI
      DO 305  K=1,2
      P(L,I,K)=PR
  305 U(L,I,K)=UR
      GO TO 315
  307 NIM=NI-1
      DELP=(PON(L)-PO1(L))/NIM
```

```
      DELU=(UON(L)-UO1(L))/NIM
      DO 309  I=1,NI
      IM=I-1
      DO 309  K=1,2
      P(L,I,K)=PO1(L)+IM*DELP
  309 U(L,I,K)=UO1(L)+IM*DELU
  315 CONTINUE
      IF(NOPT.GT.0)  CALL PTIME(TT,PSO)
C**** INITIALIZE CONSTANTS FOR PLASTIC DEFORMATION
      NPLP=0
      DO 322 KPL=1,10
  322 MPX(KPL)=0
      DO 324 L=1,NPIPE
  324 KPLAS(L)=0
      WRITE(KOUT,960)
      GO TO 700
C
C
C************************  INTERIOR NODE CALCULATION  ************************
C
  400 DO 475 L=1,NPIPE
      NI=NNODE(L)
      NIM=NI-1
      IF (NI.LE.2) GO TO 475
      IF(KPLAS(L).GT.0) GO TO 430
C**** ELASTIC PIPE DEFORMATION
      DO 410  I=2,NIM
      CALL INTRP (L,I,K1,UA,PA,CA)
      CALL FRCTRM (L,UA,GA)
      CALL INTRP (L,I,K2,UB,PB,CB)
      CALL FRCTRM (L,UB,GB)
      U(L,I,2)= (CA*(UA-DT*GA)+CB*(UB-DT*GB)+(PA-PB)/DENS)/(CA+CB)
      P(L,I,2)= (DENS*(UA-UB+DT*(GB-GA))*CA*CB+PA*CB+PB*CA)/(CA+CB)
      IF(ABS(U(L,I,2)-U(L,I,1)).LT.0.00001)  U(L,I,2)=U(L,I,1)
      IF(ABS(P(L,I,2)-P(L,I,1)).LT.0.1)  P(L,I,2)=P(L,I,1)
C     CHECK FOR PLASTIC DEFORMATION IN ELASTIC PIPE
      IF(P(L,I,2).GT.PY(L)) GO TO 420
  410 CONTINUE
      GO TO 475
C**** PIPE L BECOMES PLASTIC
  420 NPLP=NPLP+1
      IF(NPLP.GT.NPLMX) GO TO 850
      KPLAS(L)=NPLP
      KPL=KPLAS(L)
      LPLP(NPLP)=L
      WRITE(KOUT,970) LPIPE(L),NSTEP,(LPIPE(LPLP(KK)),KK=1,NPLP)
      DO 422  IP=1,NI
  422 PMX(KPL,IP)=P(L,IP,1)
C**** PLASTIC PIPE DEFORMATION
C**** PLASTIC PIPE DEFORMATION
  430 KPL=KPLAS(L)
      DO 465  I=2,NIM
      PP1=P(L,I,1)
      UP1=U(L,I,1)
      PMXQ=PMX(KPL,I)
```

```
      KIT=0
      P(L,I,2)=PP1
      U(L,I,2)=UP1
C     CALCULATION AT POINT P
  432 CALL INTRP (L,I,K1,UA,PA,CA)
      CALL FRCTRM (L,UA,GA)
      CALL INTRP (L,I,K2,UB,PB,CB)
      CALL FRCTRM (L,UB,GB)
      U(L,I,2)= (CA*(UA-DT*GA)+CB*(UB-DT*GB)+(PA-PB)/DENS)/(CA+CB)
      P(L,I,2)= (DENS*(UA-UB+DT*(GB-GA))*CA*CB+PA*CB+PB*CA)/(CA+CB)
      IF(ABS(U(L,I,2)-U(L,I,1)).LT.0.00001)  U(L,I,2)=U(L,I,1)
      IF(ABS(P(L,I,2)-P(L,I,1)).LT.0.1)  P(L,I,2)=P(L,I,1)
      PP2=P(L,I,2)
      UP2=U(L,I,2)
      IF(ABS(PP2-PP1).LE.ABS(EPSI*PP2))  GO TO 460
      IF(KIT.EQ.20) GO TO 440
      GO TO 455
C     POOR CONVERGENCE OF ITERATION
  440 MPX(KPL)=1
      CALL WVSPD(MAT(L),HOD(L),PP1,PMXQ,CP1)
      CALL WVSPD(MAT(L),HOD(L),PP2,PMXQ,CP2)
      CPX(KPL)=(CP1+CP2)/2.0
  455 KIT=KIT+1
      IF(KIT.GT.30) U(L,I,2)=(UP1+UP2)/2.0
      IF(KIT.GT.40) GO TO 855
      PP1=PP2
      UP1=U(L,I,2)
      GO TO 432
  460 MPX(KPL)=0
  465 CONTINUE
  475 CONTINUE
C
C
C************************  JUNCTION NODE CALCULATION  *************************
C
      DO 599 J=1,NOJUN
      KIT=0
      JT=JTYPE(J)
      NN1=ML1(J)
      NNN=MLN(J)
      NT=NN1+NNN
  502 CONTINUF
      IF(JT.EQ.1) GO TO 505
      IF(JT.EQ.3) GO TO 510
      IF(JT.EQ.4) GO TO 511
      IF(JT.EQ.6) GO TO 515
      IF(JT.EQ.7) GO TO 520
      IF(JT.EQ.8) GO TO 525
      IF(JT.EQ.9) GO TO 530
      IF(JT.EQ.10) GO TO 535
      IF(JT.EQ.15) GO TO 540
      GO TO 835
  505 CALL AREACH(J)
      GO TO 575
  510 CALL TEE(J,INCOND,PR)
```

```
      GO TO 575
  511 DO 512  K=1,NOPUMP
      KK=K
      IF(JPUMP(K).EQ.JUN(J)) GO TO 513
  512 CONTINUE
      GO TO 847
  513 CALL PUMP(J,LLPMP(KK),HEAD(KK))
      GO TO 575
  515 CALL IMPED(J)
      GO TO 575
  520 CALL CLOSED(J)
      GO TO 575
  525 DO 526 K=1,NOTNK
      KK=K
      IF(JTANK(K).EQ.JUN(J))  GO TO 527
  526 CONTINUE
      GO TO 843
  527 CALL CONSTP(J,PTANK(KK))
      GO TO 575
  530 CALL FAREND(J)
      GO TO 575
  535 DO 536 K=1,NORD
      KK=K
      IF(JRD(K).EQ.JUN(J))  GO TO 537
  536 CONTINUE
      GO TO 845
  537 KKRD=KRD(KK)
      CALL RUPDSK (J,PRDB(KK),PRDG(KK),KKRD)
      IF(KRD(KK).EQ.0 .AND. KKRD.EQ.1) GO TO 538
      GO TO 575
  538 WRITE(KOUT,966) JUN(J),NSTEP,TT
      TRD(KK)=II
      KRD(KK)=1
      GO TO 575
  540 CALL PRESSO(J,PSO)
      GO TO 575
  575 CONTINUE
C**** CHECK FOR PLASTIC DEFORMATION AT JUNCTION
      KPIND=0
      DO 590  K=1,NT
      IF(K.GT.NN1) GO TO 578
      L=LJ1(J,K)
      I=1
      GO TO 580
  578 KM=K-NN1
      L-LJN(J,KM)
      I=NNODE(L)
  580 KPL=KPLAS(L)
      IF(KPL.EQ.0) GO TO 586
      IF(KIT.EQ.0) PPIJ(KPL)=P(L,I,1)
      PP2=P(L,I,2)
      PP1=PPIJ(KPL)
      PPIJ(KPL)=PP2
      IF(ABS(PP2-PP1).LE.ABS(EPSI*PP2))  GO TO 590
      KPIND=1
```

```
      IF(KIT.EQ.20) GO TO 582
      GO TO 590
C     POOR CONVERGENCE AT PLASTIC JUNCTION
  582 PMXQ=PMX(KPL,I)
      CALL WVSPD(MAT(L),HOD(L),PP1,PMXQ,CP1)
      CALL WVSPD(MAT(L),HOD(L),PP2,PMXQ,CP2)
      MPX(KPL)=1
      CPX(KPL)=(CP1+CP2)/2.0
      GO TO 590
  586 IF(P(L,I,2).LE.PY(L)) GO TO 590
C**** PIPE BECOMES PLASTIC AT JUNCTION
      KPIND=1
      NPLP=NPLP+1
      IF(NPLP.GT.NPLMX) GO TO 850
      KPLAS(L)=NPLP
      KPL=KPLAS(L)
      LPLP(KPL)=L
      WRITE(KOUT,970) LPIPE(L),NSTEP,(LPIPE(LPLP(KK)),KK=1,NPLP)
      NI=NNODE(L)
      DO 588  IP=1,NI
  588 PMX(KPL,IP)=P(L,IP,1)
      PPIJ(KPL)=P(L,I,2)
  590 CONTINUE
      IF(KPIND.EQ.0) GO TO 592
      KIT=KIT+1
      IF(KIT.GT.30) U(L,I,2)=(U(L,I,2)+UPIJ(KPL))/2.0
      UPIJ(KPL)=U(L,I,2)
      IF(KIT.GT.40) GO TO 860
      GO TO 502
  592 IF(KIT.EQ.0) GO TO 599
      DO 594  KPL=1,NPLP
  594 MPX(KPL)=0
  599 CONTINUE
C
C
C******************  PRINT RESULTS AT SPECIFIED TIME STEP  *********************
C
      IF(KPRIN.EQ.NOPRIN) GO TO 700
      KPRIN=KPRIN+1
      GO TO 725
  700 KPRIN=1
      WRITE(KOUT,961)  TT,NSTEP
      IF(NOPT.EQ.0) GO TO 701
      PSOW=PSO/AFAC
      WRITE(KOUT,965) PSOW
  701 DO 710 L=1,NPIPE
      IF (NLPRN.EQ.0) GO TO 704
      DO 702   K=1,NLPRN
      IF(LPIPE(L).EQ.LPRIN(K)) GO TO 704
  702 CONTINUE
      GO TO 710
  704 WRITE(KOUT,962) LPIPE(L),J1(L),JN(L)
      NIM=NNODE(L)-1
      COUT=C(L)
      DO 706 I=1,NIM,IPRIN
```

```
      IF(KPLAS(L).GT.0) CALL WVSPD(MAT(L),HOD(L),P(L,I,2),
     X    PMX(KPLAS(L),I),COUT)
      Z=(I-1)*DZ(L)
      PPW=P(L,I,2)/AFAC
      WRITE(KOUT,963) Z,PPW,U(L,I,2),COUT
  706 CONTINUE
      Z=PLNGTH(L)
      I=NNODE(L)
      PPW=P(L,I,2)/AFAC
      IF(KPLAS(L).GT.0) CALL WVSPD(MAT(L),HOD(L),P(L,I,2),
     X    PMX(KPLAS(L),I),COUT)
      WRITE(KOUT,963) Z,PPW,U(L,I,2),COUT
  710 CONTINUE
      IF(NSTEP.EQ.0) GO TO 750
C
C
C*******************  INITIALIZATIONS FOR NEXT TIME STEP  *********************
C
C**** DETERMINE MAXIMUM PRESSURE AND VELOCITY IN EACH PIPE
  725 UMX=0.
      CUMX=0.
      DO 729 L=1,NPIPE
      NI=NNODE(L)
      PPX=PMAX(L)
      DO 727 I=1,NI
      UMX=AMAX1(UMX,ABS(U(L,I,2)))
  727 PMAX(L)=AMAX1(PMAX(L),P(L,I,2))
      IF (PMAX(L).GT.PPX) TMAX(L)=TT
      CUP=1.+UMX/C(L)
  729 CUMX=AMAX1(CUP,CUMX)
C**** REDETERMINE DT FOR NUMERICAL STABILITY
      IF (CUMX .LT. FACT1)  GO TO 735
      IF (CUMX .GT. FACT2)  GO TO 875
      FACT=1.01*CUMX
      DT=DT*FACT1/FACT
      FACT1=FACT
      FACT2=FACT2+FACT-1.
      GO TO 877
  735 CONTINUE
C**** DETERMINE PMAX IN PLASTIC PIPES
      IF(NPLP.EQ.0) GO TO 750
      DO 740 KPL=1,NPLP
      L=LPLP(KPL)
      NI=NNODE(L)
      DO 740 IP=1,NI
      PMX(KPL,IP)=AMAX1(PMX(KPL,IP),P(L,IP,2))
  740 CONTINUE
C**** INCREASE TT AND NSTEP
  750 TT-TT+DT
      IF(TT.GT.TFIN) GO TO 775
      NSTEP=NSTEP+1
C**** INITIALIZE P AND U
      DO 760 L=1,NPIPE
      NI=NNODE(L)
      DO 760 I=1,NI
```

```
      U(L,I,1)=U(L,I,2)
  760 P(L,I,1)=P(L,I,2)
C**** COMPUTE SOURCE PRESSURE
      IF(NOPT.GT.0)  CALL PTIME(TT,PSO)
      GO TO 400
C
C
C***********************  PRINTOUTS AT END OF PROBLEM  *************************
C
  775 WRITE(KOUT,975)
      WRITE(KOUT,977)
      DO 780 L=1,NPIPE
      PMM=PMAX(L)/AFAC
  780 WRITE(KOUT,978) LPIPE(L),PMM,TMAX(L)
      IF(NPLP.GT.0) WRITE(KOUT,971) (LPIPE(LPLP(K)),K=1,NPLP)
      IF(NORD.EQ.0) GO TO 799
      DO 782  K=1,NORD
      IF(KRD(K).EQ.1) WRITE(KOUT,979) JRD(K),TRD(K)
  782 CONTINUE
  799 STOP
C
C
C************************  DIAGNOSTIC MESSAGE PRINTS  ***************************
C
  800 WRITE(KOUT,801)
  801 FORMAT (/,10X,'ERROR IN NUMBER OF PIPES')
      STOP
  805 WRITE(KOUT,806)
  806 FORMAT (/,10X,'ERROR IN NUMBER OF JUNCTIONS')
      STOP
  810 WRITE(KOUT,811)
  811 FORMAT(/,10X,'ERROR IN NUMBER OF TANKS')
      STOP
  812 WRITE(KOUT,813) JUN(J)
  813 FORMAT(/,10X,'NOTNK=0 AND JUNCTION',I5,' IS A SURGE TANK')
      STOP
  815 WRITE(KOUT,816)
  816 FORMAT(/,10X,'ERROR IN NUMBER OF RUPTURE DISKS')
      STOP
  817 WRITE(KOUT,818) JUN(J)
  818 FORMAT(/,10X,'NORD=0 AND JUNCTION',I5,' IS A RUPTURE DISK')
      STOP
  820 WRITE(KOUT,821)
  821 FORMAT(/,10X,'ERROR IN NUMBER OF PUMPS')
      STOP
  822 WRITE(KOUT,823) JPUMP(K),LPUMP(K)
  823 FORMAT(/,10X,'PUMP DISCHARGE PIPE NUMBER IS INCORRECT',/,
     X    10X,'JPUMP =',I5,',  LPUMP =',I5)
      STOP
  824 WRITE(KOUT,825) JUN(J)
  825 FORMAT(/,10X,'NOPUMP=0 AND JUNCTION',I5,' IS A PUMP')
      STOP
  828 WRITE(KOUT,829)
  829 FORMAT(/,10X,'ERROR IN NUMBER OF PRESSURE SOURCE DATA POINTS')
      STOP
```

```
  830 WRITE(KOUT,831)  JUN(J),JTYPE(J)
  831 FORMAT(//10X,'INCORRECT PIPE ARRANGEMENT AT JUNCTION',
     X     I4,',  JTYPE =',I4)
      STOP
  835 WRITE(KOUT,836) JUN(J),JTYPE(J)
  836 FORMAT(//10X,'INVALID JUNCTION TYPE,  JUN =',I4,',  JTYPE =',I4)
      STOP
  837 WRITE(KOUT,838) JUN(J),JTYPE(J)
  838 FORMAT(//,10X,'PIPES CONNECTED AT JUNCTION',I4,
     X     ' HAVE DIFFERENT AREAS, JTYPE =',I4)
      STOP
  840 WRITE(KOUT,841) JUN(J)
  841 FORMAT(/,10X,'NO INPUT PRESSURE PULSE (NOPT=0) AT SOURCE',
     X     ' JUNCTION',I4)
      STOP
  843 WRITE(KOUT,844) JUN(J)
  844 FORMAT(/,10X,'NO TANK DATA INPUT FOR JUNCTION',I4)
      STOP
  845 WRITE(KOUT,846) JUN(J)
  846 FORMAT(/,10X,'NO RUPTURE DISK DATA INPUT FOR JUNCTION',I4)
      STOP
  847 WRITE(KOUT,848) JUN(J)
  848 FORMAT(/,10X,'NO PUMP DATA INPUT FOR JUNCTION',I5)
      STOP
  850 NPLP=NPLP-1
      WRITE(KOUT,970) LPIPE(L),NSTEP,(LPIPE(LPLP(KK)),KK=1,NPLP)
      WRITE(KOUT,851)
  851 FORMAT(1H1,////,10X,'NUMBER OF PLASTIC PIPES EXCEEDS NPLMX')
      GO TO 775
  855 WRITE(KOUT,856) LPIPE(L),I,TT
  856 FORMAT(1H1,////,10X,'NO CONVERGENCE IN PLASTIC ITERATION FOR',
     X     ' PIPE NO',I4,',  I =',I4,',  TIME =',E12.5,' SEC')
      GO TO 775
  860 WRITE(KOUT,861) JUN(J),TT
  861 FORMAT(1H1,////,10X,'NO CONVERGENCE IN PLASTIC ITERATION AT',
     X     ' JUNCTION',I4,',  TIME =',E12.5,' SEC')
      GO TO 775
  870 WRITE(KOUT,871)  LPIPE(L),DT,DTT
  871 FORMAT(////,10X,'NNODE .GT. IMAX FOR PIPE NO',I4,
     X     ' - RESET DT =',E12.5,' TO DT =',E12.5)
      GO TO 111
  875 WRITE(KOUT,876)  CUMX
  876 FORMAT(////,10X,'CUMX =',E12.5,' IS GREATER THAN FACT2')
      GO TO 775
  877 WRITE(KOUT,878)  DT
  878 FORMAT (5X,/,'***** DT IS DECREASED TO ',E12.5,' *****',/)
      GO TO 735
C
C
C************************  INPUT FORMAT STATEMENTS  ***************************
C
  900 FORMAT(16I5)
  901 FORMAT(8E10.5)
  905 FORMAT(4I5,5E10.5)
  910 FORMAT(I10,E10.5)
```

```
  911 FORMAT(I10,2E10.5)
  912 FORMAT(2I5,E10.5)
C
C
C***********************  OUTPUT FORMAT STATEMENTS  ****************************
C
  930 FORMAT (1H1,//,30X,'------  PROBLEM PARAMETERS AND DESCRIPTION',
     1     ' OF NETWORK ARRANGEMENT  ------',////,5X,'NPIPE  =',I5,7X,
     2     'NOJUN  =',I5,7X,'NOTNK  =',I5,7X,'NORD   =',I5,7X,'NOPUMP =',
     3     I5,7X,'KFRIC  =',I5,/,5X,'NOPT   =',I5,7X,'INCOND =',I5,
     4     7X,'NOPRIN =',I5,7X,'IPRIN  =',I5,7X,'NLPRN  =',I5,///)
  932 FORMAT(5X,'DT =',E12.5,' SEC',12X,'TBEG =',E12.5,' SEC',
     1       13X,'TFIN =',E12.5,' SEC',/,
     2     5X,'UR =',F12.4,' FT/SEC',9X,'PR =',F14.3,' PSI',13X,
     3     'TEMP =',F10.3,' DEG.F',//,5X,'SP WEIGHT =',F8.3,' LBS/FT3',
     4     5X,'VISC =',E12.5,' LBS-SEC/FT2',5X,'SOUND SPEED =',F10.2,
     5     ' FT/SEC',///)
  934 FORMAT(////,38X,'---  CONSTANT PRESSURE TANKS  ---',//,41X,
     X     'JUNCTION NO',4X,'GAS PRESSURE',/)
  935 FORMAT(44X,I5,7X,F12.3)
  937 FORMAT(////,43X,'---  RUPTURE DISKS  ---',//,35X,
     X     'JUNCTION NO',3X,'BURST PRESSURE',3X,'GAS PRESSURE',/)
  938 FORMAT(38X,I5,8X,F12.3,3X,F12.3)
  940 FORMAT(////,47X,'---  PUMPS  ---',//,32X,
     X     'JUNCTION NO',3X,'DISCHARGE PIPE NO',4X,'HEAD, PSI',/)
  941 FORMAT(35X,I5,12X,I5,8X,F12.3)
  945 FORMAT(1H1,//,6X,'PIPE',  13X,'JUNCTION NO(TYPE)',14X,
     1     'PRES URE - PSI',13X,'VELOCITY - FT/SEC',/,7X,'NO',
     2     12X,'I = 1',11X,'I = N',7X,'FIRST NODE',5X,'LAST NODE',
     3     4X,'FIRST NODE',5X,'LAST NODE',/)
  946 FORMAT (5X,I4,9X,I4,'(',I2,')',8X,I4,'(',I2,')',3X,2F14.3,2F14.4)
  950 FORMAT (1H1,//,6X,'PIPE',  9X,'DIAMETER',8X,'THICKNESS',11X,
     1     'AREA',13X,'LENGTH',9X,'INTERNODAL',7X,'NO OF',/,7X,
     2     'NO',12X,'INCH',13X,'INCH',11X,'SQR INCH',12X,'FEET',11X,
     3     'DIST, FT',8X,'NODES',/)
  951 FORMAT (5X,I4,1X,F17.4,F17.5,2F17.3,F17.4,9X,I4)
  953 FORMAT (1H1,//,6X,'PIPE',  5X,'MATERIAL',5X,'YIELD PRESSURE',5X,
     1     'EL. WAVE SPEED',8X,'ROUGHNESS',/,
     2     7X,'NO',10X,'NO',14X,'PSI',14X,'FT/SEC',14X,'INCH',/)
  954 FORMAT (5X,I4,10X,I2,5X,F17.2,2X,F17.2,2X,E17.4)
  956 FORMAT (1H1,////,40X,'-----  INPUT DATA FOR PRESSURE - TIME',
     X     ' HISTORY  -----',///,52X,'TIME, SEC',6X,'PRESSURE, PSI',/)
  957 FORMAT(50X,E12.5,5X,F12.3)
  960 FORMAT (1H1,//,40X,'------  RESULTS OF PRESSURE TRANSIENT',
     X     ' CALCULATION  ------')
  961 FORMAT(////,5X,'**** TIME =',E15.5,' SEC',5X,'(',I4,' STEPS )')
  962 FORMAT(/,3X,'PIPE NO',I5,',  J1 =',I4,',  JN =',I4,/,
     1     22X,'DISTANCE, FT',12X,'PRESSURE, PSI',11X,
     2     'VELOCITY, FT/SEC',11X,'WAVE SPD, FT/SEC')
  963 FORMAT (19X,F15.4,10X,F15.2,10X,F15.3,10X,F15.2)
  965 FORMAT(/,15X,'SOURCE PRESSURE =',F10.2,' PSI')
  966 FORMAT(////,10X,'**** RUPTURE DISK AT JUNCTION',I4,
     X     ' HAS BURST IN TIME STEP NO.',I5,',  T =',E12.5)
  970 FORMAT(////,10X,'**** PIPE NUMBER',I4,' HAS BECOME PLASTIC IN',
     1     ' TIME STEP NO.',I4,//,15X,'PIPES WHICH HAVE HAD SOME',
```

```
   2     ' PLASTIC DEFORMATION ARE NUMBERS',10I4,/,(20X,20I4,/))
971 FORMAT(////,15X,'PIPES WHICH HAVE HAD SOME',
   X     ' PLASTIC DEFORMATION ARE NUMBERS',10I4,/,(20X,20I4,/))
975 FORMAT (//,45X,'------  END OF CALCULATION  ------')
977 FORMAT (1H1,//,30X,'------  MAXIMUM PRESSURES IN EACH PIPE',
   X     ' DURING CALCULATION  ------',/)
978 FORMAT (25X,'MAXIMUM PRESSURE IN PIPE',  I5,' = ',F12.2,
   X     ' PSI OCCURRED AT ',E12.5,' SEC')
979 FORMAT(///,15X,'RUPTURE DISK AT JUNCTION',I4,' BURST AT',
   X     E12.5,' SEC')
    END
```

```
      SUBROUTINE MTPRP
C**** SUBROUTINE WHICH COMPUTES WAVE SPEED AS FUNCTION OF PRESSURE
C**** FOR VARIOUS MATERIALS
C         MAT=1 - 304 STAINLESS STEEL
C         MAT=2 - 316 STAINLESS STEEL
C         MAT=3 - LINEAR ELASTIC AND PLASTIC REGIONS WITH ELLIPTIC
C                 TRANSITION
C         MAT=4 - BILINEAR STRESS-STRAIN RELATION
C         MAT=5 - BILINEAR STRESS-STRAIN RELATION
C         MAT=6 - RIGID PIPE WALL
      COMMON  /R1/ A(100),C(100),DZ(100),D(100),P(100,100,2),
     1             U(100,100,2),LJ1(100,6),LJN(100,6),ML1(100),MLN(100),
     2             LPIPE(100),JUN(100),NNODE(100),DENS,DT
      COMMON  /R3/ TR,CO
C
      DIMENSION E(6),SY(6),KPR(6),EPSY(6)
C
      DATA  KOUT/6/,RRR/1.0001/,PFAC/144.0/,KPR/6*0/,PLAR/1.0E+06/
C
C**** COMPUTES YIELD PRESSURE AND ELASTIC WAVE SPEED
      ENTRY ELPRP(M,HD,PYLD,CELAS)
      IF(KPR(M).GT.0) GO TO 5
      GO TO (10,20,30,40,50,60),M
    1 WRITE (KOUT,2) M
    2 FORMAT(1H0,5X,'INCORRECT MATERIAL NUMBER, MAT =',I4)
      STOP
    5 PYLD=2.0*SY(M)*HD
      CELAS=CO/SQRT(1.0+CO**2*DENS/HD/E(M))
      RETURN
C**** 304 STAINLESS STEEL PROPERTIES FROM LMFBR MATERIALS HANDBOOK
C     M = 1
      DATA AS1,AS2,AS3,AS4/11.29E+03,1.23,27.5E+03,10.7/,
     1    AF1,AF2,AF3/3.0E-06,5.03E-06,0.0143/,
     2    AF4,AF5,AF6/9.3E-06,7.23E-05,0.1877/
C     COMPUTE MATERIAL PROPERTIES
   10 SY(1)=(AS1-AS2*TR)*PFAC
      SA1=(AS3-AS4*TR)*PFAC
      A11=AF1/PFAC
      A12=AF2*TR-AF3
      A13=AF4/PFAC
      A14=AF5*TR-AF6
      E(1)=SY(1)/(A11*SY(1)+A12)**2
      EPSY(1)=SY(1)/E(1)
      KPR(1)=1
      GO TO 5
C**** 316 STAINLESS STEEL PROPERTIES FROM LMFBR MATERIALS HANDBOOK
C     M = 2
      DATA BS1,BS2,BS3,BS4/10.27E+03,2.784E+06,13.373E+03,3.616E+06/,
     1    BF1,BF2,BF3/1.8E-06,3.802,3.9E-03/,
     2    BF4,BF5,BF6/8.2E-06,26.945,0.0817/
   20 SY(2)=(BS1+BS2/TR)*PFAC
      SA2=(BS3+BS4/TR)*PFAC
      A21=BF1/PFAC
```

```
      A22=BF3-BF2/TR
      A23=BF4/PFAC
      A24=-BF5/TR-BF6
      E(2)=SY(2)/(A21*SY(2)+A22)**2
      EPSY(2)=SY(2)/E(2)
      KPR(2)=1
      GO TO 5
C**** LINEAR ELASTIC AND PLASTIC REGIONS WITH ELLIPTIC TRANSITION
C     M = 3
C     DATA FOR NICKEL 200
      DATA EE3/30.0E+06/,EPL3/0.405E+06/,SO3/29.0E+03/,
     1     GMY3/0.75/,GMA3/1.25/
   30 E(3)=EE3*PFAC
      EP3=EPL3*PFAC
      SGO3=SO3*PFAC
      RM3=EP3/E(3)
      SY(3)=SGO3*GMY3
      SA3=SGO3*GMA3
      EPSY(3)=SY(3)/E(3)
      DY3=1.0-GMY3
      DA3=GMA3-1.0
      A31=DA3+RM3*DY3
      A32=DA3-RM3*DY3
      A33=DA3*GMY3-RM3*DY3*GMA3
      A34=GMA3-GMY3
      A35=DA3*DY3+RM3*(1.0-GMY3*GMA3)
      A36=DY3-DA3
      A37=GMY3**2*(1.0-RM3**2)*DA3**2
      KPR(3)=1
      GO TO 5
C**** BILINEAR STRESS-STRAIN RELATION
C     M = 4
C     BILINEAR APPROXIMATION FOR NICKEL 200
      DATA EE4/30.0E+06/,EPL4/4.05E+05/,SY4/29.0E+03/
   40 E(4)=EE4*PFAC
      EP4=EPL4*PFAC
      SY(4)=SY4*PFAC
      RM4=EP4/E(4)
      EPSY(4)=SY(4)/E(4)
      KPR(4)=1
      GO TO 5
C**** BILINEAR STRESS-STRAIN RELATION
C     M = 5
      DATA EE5/24.8E+06/,EPL5/2.48E+06/,SY5/90.0E+03/
   50 E(5)=EE5*PFAC
      EP5=EPL5*PFAC
      SY(5)=SY5*PFAC
      RM5=EP5/E(5)
      EPSY(5)=SY(5)/E(5)
      KPR(5)=1
      GO TO 5
C**** RIGID PIPE WALL
C     M = 6
   60 PYLD=PLAR*PFAC
      CELAS=CO
```

```
      EPSY(6)=0.0
      RETURN
C
C**** PRINTOUTS DESCRIBING MATERIALS
      ENTRY WRMAT(M)
      GO TO(210,220,230,240,250,260),M
      GO TO 1
C     M = 1
  210 EOUT=E(1)/PFAC
      SYOUT=SY(1)/PFAC
      WRITE(KOUT,212) TR,EOUT,SYOUT
  212 FORMAT(////,5X,'MATERIAL 1 IS 304 STAINLESS STEEL AT',F12.3,
     1' DEG. F',/,10X,'ELASTIC MODULUS =',E13.6,' PSI,  YIELD STRESS =',
     2E13.6,' PSI')
      RETURN
C     M = 2
  220 EOUT=E(2)/PFAC
      SYOUT=SY(2)/PFAC
      WRITE(KOUT,222) TR,EOUT,SYOUT
  222 FORMAT(////,5X,'MATERIAL 2 IS 316 STAINLESS STEEL AT',F12.3,
     1' DEG. F',/,10X,'ELASTIC MODULUS =',E13.6,' PSI,  YIELD STRESS =',
     2E13.6,' PSI')
      RETURN
C     M = 3
  230 EOUT=E(3)/PFAC
      SYOUT=SY(3)/PFAC
      WRITE(KOUT,232) EOUT,SYOUT
  232 FORMAT(////,5X,'MATERIAL 3 IS NICKEL 200, USING FITTED CURVE',
     1' FOR STRESS-STRAIN RELATION',/,10X,'ELASTIC MODULUS =',E13.6,
     2' PSI,  YIELD STRESS =',E13.6,' PSI')
      RETURN
C     M = 4
  240 EOUT=E(4)/PFAC
      SYOUT=SY(4)/PFAC
      EPOUT=EP4/PFAC
      WRITE(KOUT,242) EOUT,SYOUT,EPOUT
  242 FORMAT(////,5X,'MATERIAL 4 IS BILINEAR APPROXIMATION TO NICKEL',
     1' 200',/,10X,'ELASTIC MODULUS =',E13.6,' PSI,  YIELD STRESS =',
     2E13.6,' PSI,  PLASTIC MODULUS =',E13.6,' PSI')
      RETURN
C     M = 5
  250 EOUT=E(5)/PFAC
      SYOUT=SY(5)/PFAC
      EPOUT=EP5/PFAC
      WRITE(KOUT,252) EOUT,SYOUT,EPOUT
  252 FORMAT(////,5X,'MATERIAL 5 HAS A BILINEAR STRESS-STRAIN RELATION',
     1/,10X,'ELASTIC MODULUS =',E13.6,' PSI,  YIELD STRESS =',E13.6,
     2' PSI,  PLASTIC MODULUS =',E13.6,' PSI')
      RETURN
C     M = 6
  260 WRITE(KOUT,262)
  262 FORMAT(////,5X,'MATERIAL 6 IS RIGID',/,10X,'ELASTIC MODULUS AND',
     1' YIELD STRESS ARE INFINITE',/,10X,'WAVE SPEED IS EQUAL TO ',
     2'SOUND SPEED')
      RETURN
```

```
      ENTRY WVSPD(M,HD,PP,PX,CC)
C**** CALCULATION OF WAVE SPEED AS FUNCTION OF PRESSURE AND PREVIOUS
C**** MAXIMUM PRESSURE
      SS=PP/HD/2.0
      SX=PX/HD/2.0
      BC=CO**2*DENS/HD
      GO TO (510,520,530,540,550,560),M
      GO TO 1
C**** M = 1
  510 IF(SS.GT.SX/RRR.AND.SS.GT.SY(1)) GO TO 512
C     DEFORMATION IS ELASTIC
      EEE=E(1)
      GO TO 518
C     DEFORMATION IS PLASTIC
  512 IF(SS.GT.SA1) GO TO 514
C     STRESS IS BETWEEN SY AND SA
      EEE=0.5/A11/(A11*SS+A12)
      GO TO 518
C     STRESS IS GREATER THAN SA
  514 EEE=0.5/A13/(A13*SS+A14)
  518 CC=CO/SQRT(1.0+BC/(EEE-2.0*SS))
      RETURN
C
C**** M = 2
  520 IF(SS.GT.SX/RRR.AND.SS.GT.SY(2)) GO TO 522
C     DEFORMATION IS ELASTIC
      EEE=E(2)
      GO TO 528
C     DEFORMATION IS PLASTIC
  522 IF(SS.GT.SA2) GO TO 524
C     STRESS IS BETWEEN SY AND SA
      EEE=0.5/A21/(A21*SS+A22)
      GO TO 528
C     STRESS IS GREATER THAN SA
  524 EEE=0.5/A23/(A23*SS+A24)
  528 CC=CO/SQRT(1.0+BC/(EEE-2.0*SS))
      RETURN
C
C**** M = 3
  530 IF(SS.GT.SX/RRR.AND.SS.GT.SY(3)) GO TO 532
C     DEFORMATION IS ELASTIC
      EEE=E(3)
      GO TO 538
C     DEFORMATION IS PLASTIC
  532 IF(SS.LT.SA3) GO TO 534
C     STRESS IS IN LINEAR PLASTIC REGION
      EEE=EP3
      GO TO 538
C     STRESS IS IN ELLIPTIC TRANSITION REGION
  534 SRAT=SS/SG03
      ARG=1.0-(A37+A31*(A32*SRAT**2-2.0*A33*SRAT))*A36/A34/A35**2
      EEE=EP3*A34*A35*SQRT(ARG)/A31/(A32*SRAT-A33)
  538 CC=CO/SQRT(1.0+BC/(EEE-2.0*SS))
      RETURN
C
```

```
C**** M = 4
  540 IF(SS.GT.SX/RRR.AND.SS.GT.SY(4)) GO TO 542
C      DEFORMATION IS ELASTIC
       EEE=E(4)
       GO TO 548
C      DEFORMATION IS PLASTIC
  542 EEE=EP4
  548 CC=CO/SQRT(1.0+BC/(EEE-2.0*SS))
       RETURN
C
C**** M = 5
  550 IF(SS.GT.SX/RRR.AND.SS.GT.SY(5)) GO TO 552
C      DEFORMATION IS ELASTIC
       EEE=E(5)
       GO TO 558
C      DEFORMATION IS PLASTIC
  552 EEE=EP5
  558 CC=CO/SQRT(1.0+BC/(EEE-2.0*SS))
       RETURN
C
C**** M = 6
  560 CC=CO
       RETURN
       END
```

```
      SUBROUTINE FLPRP(SPWT,DENS,VISC,SNDSPD)
C**** DETERMINES FLUID PROPERTIES AT TEMPERATURE TR, DEG.F
C**** SPWT - SPECIFIC WEIGHT, LBS/FT3
C**** DENS - DENSITY, LBS-SEC2/FT4
C**** VISC - VISCOSITY, LBS-SEC/FT2
C**** SNDSPD - SOUND SPEED, FT/SEC
C
C**** SODIUM PROPERTIES
C
      COMMON  /R3/ TR,CO
      DATA  DL0/59.566/,DL1/-0.79504E-02/,DL2/-0.2872E-06/,
     1    DL3/0.60350E-10/,CL1/8285./,CL2/-2187./,CL3/2290./,GC/32.173/
      SPWT=DL0+DL1*TR+DL2*TR**2+DL3*TR**3
      DENS=SPWT/GC
      AMU=EXP(2.303*(1.0203+397.17/(TR+460.)-0.4925*ALOG10(TR+460.)))/
     1    3600.0
      VISC=AMU/GC
      SNDSPD=CL1+CL2*(TR-210.)/CL3
      RETURN
      END
```

```
      SUBROUTINE INTRP(L,I,K,UC,PC,CC)
C**** INTERPOLATION FOR PRESSURE, VELOCITY, AND WAVE SPEED
      COMMON  /R1/ A(100),C(100),DZ(100),D(100),P(100,100,2),
     1             U(100,100,2),LJ1(100,6),LJN(100,6),ML1(100),MLN(100),
     2             LPIPE(100),JUN(100),NNODE(100),DENS,DT
      COMMON  /R4/ PMX(10,100),KPLAS(100),HOD(100),MAT(100),PY(100),
     1             CPX(10),MPX(10)
      DATA  KOUT/6/,EPSI/0.0001/
      THTA=DT/DZ(L)
      PQ=P(L,I,1)
      UQ=U(L,I,1)
      KPL=KPLAS(L)
      M=MAT(L)
      HD=HOD(L)
      IF(K.EQ.2) GO TO 10
      UR=U(L,I-1,1)
      PR=P(L,I-1,1)
      IF(KPL) 20,20,100
   10 US=U(L,I+1,1)
      PS=P(L,I+1,1)
      IF(KPL) 30,30,200
C**** ELASTIC RESPONSE
   20 CC=C(L)
      UC=(UQ-CC*THTA*(UQ-UR))/(1.0+THTA*(UQ-UR))
      PC=PQ-THTA*(CC+UC)*(PQ-PR)
      RETURN
   30 CC=C(L)
      UC=(UQ+CC*THTA*(US-UQ))/(1.0+THTA*(US-UQ))
      PC=PQ+THTA*(CC-UC)*(PS-PQ)
      RETURN
C**** PLASTIC RESPONSE
C     ITERATION AT POINT A
  100 PMXQ=PMX(KPL,I)
      PMXR=PMX(KPL,I-1)
      PP=P(L,I,2)
      UP=U(L,I,2)
      CALL WVSPD(M,HD,PQ,PMXQ,CQ)
      CALL WVSPD(M,HD,PR,PMXR,CR)
      CALL WVSPD(M,HD,PP,PMXQ,CP)
      IF(MPX(KPL).GT.0) CP=CPX(KPL)
      CA=CR
      VPL=(CP+CR+UP+UR)/2.0
      KCA=0
  110 KCA=KCA+1
      IF(KCA.GT.20) GO TO 180
      VPL1=VPL
      UA=UQ-THTA*VPL*(UQ-UR)
      PA=PQ-THTA*VPL*(PQ-PR)
      IF(ABS((CR-CQ)/CR).LT.EPSI) GO TO 113
      IF(KCA.GT.10) GO TO 113
      CA1=CA
      PMXA=PMXQ-THTA*VPL*(PMXQ-PMXR)
      CALL WVSPD(M,HD,PA,PMXA,CA)
```

```
      IF(KCA.EQ.10) CA=(CA+CA1)/2.0
  113 VPL=(CA+CP+UA+UP)/2.0
      IF(ABS((VPL-VPL1)/VPL).LT.EPSI) GO TO 114
      GO TO 110
  114 PC=PA
      UC=UA
      CC=(CA+CP)/2.0
      RETURN
  180 WRITE(KOUT,181) LPIPE(L),I,K
  181 FORMAT(1H1,10X,'NO CONVERGENCE IN INTRP ITERATION',/,10X,
     X     'LPIPE =',I4,',   I =',I4,',   K =',I2)
      STOP
C     ITERATION AT POINT B
  200 PMXQ=PMX(KPL,I)
      PMXS=PMX(KPL,I+1)
      PP=P(L,I,2)
      UP=U(L,I,2)
      CALL WVSPD(M,HD,PQ,PMXQ,CQ)
      CALL WVSPD(M,HD,PS,PMXS,CS)
      CALL WVSPD(M,HD,PP,PMXQ,CP)
      IF(MPX(KPL).GT.0) CP=CPX(KPL)
      CB=CS
      VMI=(US+UP-CS-CP)/2.0
      KCB=0
  210 KCB=KCB+1
      IF(KCB.GT.20) GO TO 180
      VMI1=VMI
      UB=UQ-THTA*VMI*(US-UQ)
      PB=PQ-THTA*VMI*(PS-PQ)
      IF(ABS((CS-CQ)/CS).LT.EPSI) GO TO 213
      IF(KCB.GT.10) GO TO 213
      CB1=CB
      PMXB=PMXQ-THTA*VMI*(PMXS-PMXQ)
      CALL WVSPD(M,HD,PB,PMXB,CB)
      IF(KCB.EQ.10) CB=(CB+CB1)/2.0
  213 VMI=(UB+UP-CB-CP)/2.0
      IF(ABS((VMI-VMI1)/VMI).LT.EPSI) GO TO 214
      GO TO 210
  214 PC=PB
      UC=UB
      CC=(CB+CP)/2.0
      RETURN
      END
```

```
      SUBROUTINE FRCTRM (L,V,G)
C**** SUBROUTINE WHICH COMPUTES 'WEISBACH' FRICTION FACTOR F USING
C**** 'COLEBROOK-WHITE' CORRELATION AND RETURNS FRICTION TERM G,
C****      G = F*V*ABS(V)/2/D
      COMMON  /R1/  A(100),C(100),DZ(100),D(100),P(100,100,2),
     1              U(100,100,2),LJ1(100,6),LJN(100,6),ML1(100),MLN(100),
     2              LPIPE(100),JUN(100),NNODE(100),DENS,DT
      COMMON  /R2/  RF(100),VISC,KFRIC
      DATA    KOUT/6/
      N=0
      F=0.
      G=0.
      Q=ABS(V)
C**** IF KFRIC=0, FRICTION IS NEGLECTED ****************************************
      IF(KFRIC.EQ.0) RETURN
      RR=RF(L)/D(L)
      RE=DENS*Q*D(L)/VISC
      IF (RE.LE.0.1) RETURN
      IF (RE.GT.3000.) GO TO 5
C**** LAMINAR FLOW ************************************************************
      F=64./RE
      GO TO 70
    5 IF(RR.GT.1.E-4) GO TO 10
C**** INITIAL ESTIMATE FOR SMOOTH PIPE BY 'BLASIUS' EQUATION ******************
      F=0.316/RE**.25
      GO TO 25
C**** INITIAL ESTIMATE FOR ROUGH PIPE BY 'VON KARMAN' EQUATION ****************
   10 F=1./(-2.*ALOG10(RR)+1.14)**2
C**** F IS DETERMINED ITERATIVELY BY 'COLEBROOK-WHITE' CORRELATION ************
   25 SF=SQRT(F)
      GO TO 35
   30 SF=FF
   35 DD=1.14-2.*ALOG10(RR+9.35/(RE*SF))
      FF=1./DD
      IF ((ABS(FF-SF)/SF).LE.0.01) GO TO 50
      IF (N.GE.50) GO TO 45
      N=N+1
      GO TO 30
   45 WRITE (KOUT,100)  LPIPE(L),V
  100 FORMAT(////,10X,'NO CONVERGENCE IN FRICTION TERM FOR LPIPE =',
     X     I4,',  VELOCITY =',E12.5,' FT/SEC')
   50 F=FF**2
   70 G=F*V*Q/2./D(L)
      RETURN
      END
```

```
      SUBROUTINE AREACH(J)
C**** SUDDEN EXPANSION OR CONTRACTION ****
      COMMON  /R1/ A(100),C(100),DZ(100),D(100),P(100,100,2),
     1             U(100,100,2),LJ1(100,6),LJN(100,6),ML1(100),MLN(100),
     2             LPIPE(100),JUN(100),NNODE(100),DENS,DT
      DATA  KOUT,K1,K2/6,1,2/
      IF(ML1(J).EQ.0) GO TO 2
C**** AT LEAST ONE PIPE IS CONNECTED AT FIRST NODE
      L1=LJ1(J,1)
      I1=1
      S1=1.0
      CALL INTRP(L1,I1,K2,UB,PB,CB)
      U1=U(L1,2,1)
      IF(ML1(J).EQ.2) GO TO 4
C**** AT LEAST ONE PIPE IS CONNECTED AT LAST NODE
    2 LN=LJN(J,1)
      IN=NNODE(LN)
      SN=1.0
      UN=U(LN,IN-1,1)
      CALL INTRP(LN,IN,K1,UA,PA,CA)
      IF(MLN(J).EQ.1) GO TO 6
C**** BOTH PIPES ARE CONNECTED AT LAST NODE
      L1=LJN(J,2)
      I1=NNODE(L1)
      S1=-1.0
      U1=-U(L1,I1-1,1)
      CALL INTRP(L1,I1,K1,UBM,PB,CB)
      UB=-UBM
      GO TO 6
C**** BOTH PIPES ARE CONNECTED AT FIRST NODE
    4 LN=LJ1(J,2)
      IN=1
      SN=-1.0
      UN=-U(LN,2,1)
      CALL INTRP(LN,IN,K2,UAM,PA,CA)
      UA=-UAM
C**** JUNCTION CALCULATION
    6 R=A(LN)/A(L1)
      FK=0.0
      UU=(R*UN+U1)/2.0
      IF(R.NE.1.0) GO TO 8
C**** NO AREA DISCONTINUITY
      DD=0.0
      GO TO 20
C**** EXPANSION CASE
    8 IF (R.LT.1.0.AND.UU.GT.0.) FK=(1.-R)**2
      IF (R.GT.1.0.AND.UU.LT.0.) FK=-(R-1.)**2
C**** CONTRACTION CASE
      IF (R.GT.1.0.AND.UU.GT.0.) FK=.45*R*(R-1.)
      IF (R.LT.1.0.AND.UU.LT.0.) FK=-.45*(1.-R)
      DD=1.-R**2-FK
   20 CALL FRCTRM(LN,UA,GA)
      CALL FRCTRM(L1,UB,GB)
```

```
      BB=CA+CB*R
      CC=2.*(CA*UA+CB*UB+(PA-PB)/DENS-DT*(CA*GA+CB*GB))
      CARG=BB**2-CC*DD
      IF(CARG.LT.0.0) GO TO 55
      UP=CC/(BB+SQRT(CARG))
      U(LN,IN,2)=SN*UP
      U(L1,I1,2)=S1*UP*R
      P(LN,IN,2)=PA-DENS*CA*(UP-UA+DT*GA)
      P(L1,I1,2)=PB+DENS*CB*(UP*R-UB+DT*GB)
      RETURN
   55 WRITE(KOUT,56)  JUN(J)
   56 FORMAT(1H0,5X,'AREACH SUBROUTINE - SQUARE ROOT HAS NEGATIVE',
     X     ' ARGUMENT AT JUNCTION',I4)
      STOP
      END
```

```
      SUBROUTINE TEE(J,IC,PR)
C**** TEE JUNCTION JOINING UP TO 6 BRANCHES ****
      COMMON  /R1/ A(100),C(100),DZ(100),D(100),P(100,100,2),
     1             U(100,100,2),LJ1(100,6),LJN(100,6),ML1(100),MLN(100),
     2             LPIPE(100),JUN(100),NNODE(100),DENS,DT
      DIMENSION UA(6),UB(6),PA(6),PB(6),CA(6),CB(6),GA(6),GB(6)
      DATA  K1,K2/1,2/
      NN1=ML1(J)
      NNN=MLN(J)
      NT=NN1+NNN
      SUM1=0.0
      SUM2=0.0
      DO 10 K=1,NT
      IF (K.GT.NN1) GO TO 5
      L=LJ1(J,K)
      CALL  INTRP (L,K1,K2,UB(K),PB(K),CB(K))
      CALL  FRCTRM (L,UB(K),GB(K))
      SUM1=SUM1+A(L)*(DENS*(-UB(K)+DT*GB(K))+PB(K)/CB(K))
      SUM2=SUM2+A(L)/CB(K)
      GO TO 10
    5 M=K-NN1
      L=LJN(J,M)
      CALL  INTRP(L,NNODE(L),K1,UA(M),PA(M),CA(M))
      CALL  FRCTRM (L,UA(M),GA(M))
      SUM1=SUM1+A(L)*(DENS*(UA(M)-DT*GA(M))+PA(M)/CA(M))
      SUM2=SUM2+A(L)/CA(M)
   10 CONTINUE
      IF (IC.EQ.1) GO TO 13
C**** IF U=0 AND P=PR IN EVERY BRANCH, SET U=0 AND PP=PR
      DO 12 K=1,NT
      IF (K.GT.NN1) GO TO 11
      IF (UB(K).EQ.0.0.AND.PB(K).EQ.PR) GO TO 12
      GO TO 13
   11 M=K-NN1
      IF (UA(M).EQ.0.0.AND.PA(M).EQ.PR) GO TO 12
      GO TO 13
   12 CONTINUE
      PP=PR
      GO TO 14
   13 PP=SUM1/SUM2
   14 DO 20 K=1,NT
      IF (K.GT.NN1) GO TO 15
      L=LJ1(J,K)
      U(L,1,2)=UB(K)+(PP-PB(K))/DENS/CB(K)-DT*GB(K)
      P(L,1,2)=PP
      GO TO 20
   15 M=K-NN1
      L=LJN(J,M)
      NL=NNODE(L)
      U(L,NL,2)=UA(M)-(PP-PA(M))/DENS/CA(M)-DT*GA(M)
      P(L,NL,2)=PP
   20 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE PUMP(J,LLPMP,HEAD)
C**** PUMP JUNCTION  -  TREATED AS TEE WITH HEAD ADDED TO PRESSURE
C**** IN PIPE LLPMP
      COMMON  /R1/ A(100),C(100),DZ(100),D(100),P(100,100,2),
     1              U(100,100,2),LJ1(100,6),LJN(100,6),ML1(100),MLN(100),
     2              LPIPE(100),JUN(100),NNODE(100),DENS,DT
      DIMENSION UA(6),UB(6),PA(6),PB(6),CA(6),CB(6),GA(6),GB(6)
      DATA  K1,K2/1,2/
      NN1=ML1(J)
      NNN=MLN(J)
      NT=NN1+NNN
      SUM1=0.0
      SUM2=0.0
      SUP=HEAD*A(LLPMP)/C(LLPMP)
      DO 10 K=1,NT
      IF (K.GT.NN1) GO TO 5
      L=LJ1(J,K)
      CALL  INTRP  (L,K1,K2,UB(K),PB(K),CB(K))
      CALL  FRCTRM  (L,UB(K),GB(K))
      SUM1=SUM1+A(L)*(DENS*(-UB(K)+DT*GB(K))+PB(K)/CB(K))
      SUM2=SUM2+A(L)/CB(K)
      GO TO 10
    5 M=K-NN1
      L=LJN(J,M)
      CALL  INTRP(L,NNODE(L),K1,UA(M),PA(M),CA(M))
      CALL  FRCTRM  (L,UA(M),GA(M))
      SUM1=SUM1+A(L)*(DENS*(UA(M)-DT*GA(M))+PA(M)/CA(M))
      SUM2=SUM2+A(L)/CA(M)
   10 CONTINUE
      PP=(SUM1-SUP)/SUM2
      DO 20 K=1,NT
      PJC=PP
      IF (K.GT.NN1) GO TO 15
      L=LJ1(J,K)
      IF(L.EQ.LLPMP)  PJC=PJC+HEAD
      U(L,1,2)=UB(K)+(PJC-PB(K))/DENS/CB(K)-DT*GB(K)
      P(L,1,2)=PJC
      GO TO 20
   15 M=K-NN1
      L=LJN(J,M)
      NL=NNODE(L)
      IF(L.EQ.LLPMP)  PJC=PJC+HEAD
      U(L,NL,2)=UA(M)-(PJC-PA(M))/DENS/CA(M)-DT*GA(M)
      P(L,NL,2)=PJC
   20 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE IMPED(J)
C**** DUMMY JUNCTION AND ACOUSTIC IMPEDANCE DISCONTINUITY ****
      COMMON  /R1/ A(100),C(100),DZ(100),D(100),P(100,100,2),
     1             U(100,100,2),LJ1(100,6),LJN(100,6),ML1(100),MLN(100),
     2             LPIPE(100),JUN(100),NNODE(100),DENS,DT
      DATA  K1,K2/1,2/
      IF(ML1(J).EQ.0) GO TO 10
C**** AT LEAST ONE PIPE IS CONNECTED AT FIRST NODE
      L1=LJ1(J,1)
      I1=1
      S1=1.0
      CALL INTRP(L1,I1,K2,UB,PB,CB)
      IF(ML1(J).EQ.2) GO TO 25
C**** AT LEAST ONE PIPE IS CONNECTED AT LAST NODE
   10 LN=LJN(J,1)
      IN=NNODE(LN)
      SN=1.0
      CALL INTRP(LN,IN,K1,UA,PA,CA)
      IF(MLN(J).EQ.1) GO TO 50
C**** BOTH PIPES ARE CONNECTED AT LAST NODE
      L1=LJN(J,2)
      I1=NNODE(L1)
      S1=-1.0
      CALL INTRP(L1,I1,K1,UBM,PB,CB)
      UB=-UBM
      GO TO 50
C**** BOTH PIPES ARE CONNECTED AT FIRST NODE
   25 LN=LJ1(J,2)
      IN=1
      SN=-1.0
      CALL INTRP(LN,IN,K2,UAM,PA,CA)
      UA=-UAM
C**** JUNCTION CALCULATION
   50 CALL FRCTRM(L1,UB,GB)
      CALL FRCTRM(LN,UA,GA)
      P(LN,IN,2)=(DENS*(UA-UB+DT*(GB-GA))*CA*CB+PA*CB+PB*CA)/(CA+CB)
      P(L1,I1,2)=P(LN,IN,2)
      UP=(CA*(UA-DT*GA)+CB*(UB-DT*GB)+(PA-PB)/DENS)/(CA+CB)
      U(LN,IN,2)=SN*UP
      U(L1,I1,2)=S1*UP
      RETURN
      END
```

```
      SUBROUTINE CLOSED(J)
C**** CLOSED END JUNCTIONS ****
      COMMON  /R1/ A(100),C(100),DZ(100),D(100),P(100,100,2),
     1             U(100,100,2),LJ1(100,6),LJN(100,6),ML1(100),MLN(100),
     2             LPIPE(100),JUN(100),NNODE(100),DENS,DT
      DATA  K1,K2/1,2/
      IF(MLN(J).EQ.1) GO TO 10
C**** FIRST PIPE NODE CONNECTED TO JUNCTION ****
      L=LJ1(J,1)
      CALL INTRP(L,K1,K2,UB,PB,CB)
      CALL FRCTRM(L,UB,GB)
      U(L,1,2)=0.0
      P(L,1,2)=PB+DENS*CB*(-UB+DT*GB)
      RETURN
C**** LAST PIPE NODE CONNECTED TO JUNCTION ****
   10 L=LJN(J,1)
      I=NNODE(L)
      CALL INTRP(L,I,K1,UA,PA,CA)
      CALL FRCTRM(L,UA,GA)
      U(L,I,2)=0.0
      P(L,I,2)=PA+DENS*CA*(UA-DT*GA)
      RETURN
      END
```

```
      SUBROUTINE CONSTP(J,PRES)
C**** CONSTANT PRESSURE JUNCTIONS AND RESERVOIRS ********
      COMMON  /R1/ A(100),C(100),DZ(100),D(100),P(100,100,2),
     1             U(100,100,2),LJ1(100,6),LJN(100,6),ML1(100),MLN(100),
     2             LPIPE(100),JUN(100),NNODE(100),DENS,DT
      DATA K1,K2/1,2/
      IF(ML1(J).EQ.1) GO TO 10
C**** LAST PIPE NODE CONNECTED TO JUNCTION ****
      L=LJN(J,1)
      I=NNODE(L)
      P(L,I,2)=PRES
      CALL INTRP(L,I,K1,UA,PA,CA)
      CALL FRCTRM(L,UA,GA)
      U(L,I,2)=UA-(PRES-PA)/DENS/CA-DT*GA
      RETURN
C**** FIRST PIPE NODE CONNECTED TO JUNCTION ****
   10 L=LJ1(J,1)
      P(L,1,2)=PRES
      CALL INTRP(I,K1,K2,UB,PB,CB)
      CALL FRCTRM(L,UB,GB)
      U(L,1,2)=UB+(PRES-PB)/DENS/CB-DT*GB
      RETURN
      END
```

```
      SUBROUTINE FAREND(J)
C**** NONREFLECTING FAR END JUNCTION ****
      COMMON  /R1/ A(100),C(100),DZ(100),D(100),P(100,100,2),
     1             U(100,100,2),LJ1(100,6),LJN(100,6),ML1(100),MLN(100),
     2             LPIPE(100),JUN(100),NNODE(100),DENS,DT
      IF(ML1(J).EQ.1) GO TO 10
C**** LAST PIPE NODE CONNECTED TO JUNCTION ****
      L=LJN(J,1)
      I=NNODE(L)
      P(L,I,2)=P(L,I-1,2)
      U(L,I,2)=U(L,I-1,2)
      RETURN
C**** FIRST PIPE NODE CONNECTED TO JUNCTION ****
   10 L=LJ1(J,1)
      P(L,1,2)=P(L,2,2)
      U(L,1,2)=U(L,2,2)
      RETURN
      END
```

```
      SUBROUTINE RUPDSK(J,PBUR,PRES,KRD)
C**** RUPTURE DISK SUBROUTINE ****
C**** TREATED AS A CLOSED END UNTIL PRESSURE EXCEEDS PBUR ****
C**** TREATED AS A RESERVOIR AT PRES THEREAFTER ****
      COMMON  /R1/ A(100),C(100),DZ(100),D(100),P(100,100,2),
     1             U(100,100,2),LJ1(100,6),LJN(100,6),ML1(100),MLN(100),
     2             LPIPE(100),JUN(100),NNODE(100),DENS,DT
      DATA K1,K2/1,2/
      IF(ML1(J).EQ.1) GO TO 50
C**** LAST PIPE NODE CONNECTED TO JUNCTION ****
      L=LJN(J,1)
      I=NNODE(L)
      IF(KRD.EQ.1) GO TO 10
C**** DISK HAS NOT BEEN RUPTURED PREVIOUSLY ****
      CALL INTRP(L,I,K1,UA,PA,CA)
      CALL FRCTRM(L,UA,GA)
      U(L,I,2)=0.0
      P(L,I,2)=PA+DENS*CA*(UA-DT*GA)
      IF(P(L,I,2).GT.PBUR) GO TO 8
      RETURN
C**** DISK HAS RUPTURED ****
    8 KRD=1
   10 P(L,I,2)=PRES
      CALL INTRP(L,I,K1,UA,PA,CA)
      CALL FRCTRM(L,UA,GA)
      U(L,I,2)=UA-(PRES-PA)/DENS/CA-DT*GA
      RETURN
C**** FIRST PIPE NODE CONNECTED TO JUNCTION ****
   50 L=LJ1(J,1)
      IF(KRD.EQ.1) GO TO 60
C**** DISK HAS NOT BEEN RUPTURED PREVIOUSLY ****
      CALL INTRP(L,K1,K2,UB,PB,CB)
      CALL FRCTRM(L,UB,GB)
      U(L,1,2)=0.0
      P(L,1,2)=PB+DENS*CB*(-UB+DT*GB)
      IF(P(L,1,2).GT.PBUR) GO TO 58
      RETURN
C**** DISK HAS RUPTURED ****
   58 KRD=1
   60 P(L,1,2)=PRES
      CALL INTRP(L,K1,K2,UB,PB,CB)
      CALL FRCTRM(L,UB,GB)
      U(L,1,2)=UB+(PRES-PB)/DENS/CB-DT*GB
      RETURN
      END
```

```
      SUBROUTINE PRESSO(JT,PSO)
C**** COMPUTATION OF PRESSURE SOURCE JUNCTION ****
      COMMON  /R1/  A(100),C(100),DZ(100),D(100),P(100,100,2),
     1              U(100,100,2),LJ1(100,6),LJN(100,6),ML1(100),MLN(100),
     2              LPIPE(100),JUN(100),NNODE(100),DENS,DT
      DATA  K1,K2/1,2/
      NN1=ML1(JT)
      NNN=MLN(JT)
      IF(NN1.EQ.0) GO TO 25
C**** COMPUTE PIPES JOINED WITH FIRST NODE ****
      DO 20 K=1,NN1
      L=LJ1(JT,K)
      CALL INTRP(L,K1,K2,UB,PB,CB)
      CALL FRCTRM(L,UB,GB)
      U(L,1,2)=UB+(PSO-PB)/DENS/CB-DT*GB
   20 P(L,1,2)=PSO
   25 IF(NNN.EQ.0) GO TO 35
C**** COMPUTE PIPES JOINED WITH LAST NODE ****
      DO 30 K=1,NNN
      L=LJN(JT,K)
      I=NNODE(L)
      CALL INTRP(L,I,K1,UA,PA,CA)
      CALL FRCTRM(L,UA,GA)
      U(L,I,2)=UA-(PSO-PA)/DENS/CA-DT*GA
   30 P(L,I,2)=PSO
   35 RETURN
      END
```

```
      SUBROUTINE PTIME(T,P)
C**** CALCULATES SOURCE PRESSURE P AT TIME T BY LINEAR INTERPOLATION
C**** OF VALUES IN COMMON
C
      COMMON  /R5/ NOPT,TIME(50),PTM(50)
      IF(T.LE.TIME(1)) GO TO 30
      DO 10 K=2,NOPT
      IF(T.EQ.TIME(K)) GO TO 15
      IF(T.LT.TIME(K)) GO TO 20
   10 CONTINUE
      P=PTM(NOPT)
      RETURN
   15 P=PTM(K)
      RETURN
   20 KM=K-1
      P=PTM(KM)+(PTM(K)-PTM(KM))*(T-TIME(KM))/(TIME(K)-TIME(KM))
      RETURN
   30 P=PTM(1)
      RETURN
      END
```

ACKNOWLEDGMENT

We wish to thank Dr. Y. W. Shin of the Components Technology Division for his helpful discussions and contributions to the development of PTA-1.

REFERENCES

1. G. L. Fox, Jr. and D. D. Stepnewski, *Pressure Wave Transmission in a Fluid Contained in a Plastically Deforming Pipe*, Trans. ASME, J. Pressure Vessel Technol. *96*(4), 258-262 (Nov 1974).
2. A. L. Florence and G. R. Abrahamson, *Simulation of a Hypothetical Core Disruptive Accident in a Fast Flux Test Facility*, HEDL-SRI-1 (May 1973).
3. Y. W. Shin and W. L. Chen, *Numerical Fluid-Hammer Analysis by the Method of Characteristics in Complex Piping Networks*, Nucl. Eng. Des. *33*, 357-369 (1975).
4. Y. W. Shin, *The Method of Characteristics for Analysis of Pressure Transients Resulting from Sodium-Water Reaction in Hydraulic Networks*, ANL-8049 (Dec 1973).
5. C. K. Youngdahl and C. A. Kot, *Effect of Plastic Deformation of Piping on Fluid-Transient Propagation*, Nucl. Eng. Des. *35*, 315-325 (1975).
6. C. K. Youngdahl and C. A. Kot, *Computation of the Effect of Pipe Plasticity on Pressure-pulse Propagation in a Fluid System*, ANL-75-5 (Apr 1975).
7. V. L. Streeter and E. B. Wylie, *Hydraulic Transients*, McGraw-Hill Book Co., New York (1967).
8. T. A. Zaker and M. A. Salmon, *Effects of Tube Rupture in Sodium Heated Steam Generator Units*, ASME Publication, 69-WA/NE-18 (1969).
9. R. Courant, K. Friedrichs, and H. Lewy, *Uber die Partiellen Differenzen-gleichungen der Mathematischen Physik*, Math. Ann. *100*, 32-74 (1928).
10. R. L. Daugherty and A. C. Ingersoll, *Fluid Mechanics with Engineering Applications*, McGraw-Hill Book Co., New York (1954).
11. G. H. Golden and J. V. Tokar, *Thermophysical Properties of Sodium*, ANL-7323 (Aug 1967).
12. *Nuclear Systems Material Handbook*, TID-26666 (1975).

Distribution of ANL-76-64

Internal:

J. A. Kyger
A. Amorosi
R. Avery
L. Burris
S. A. Davis
B. R. T. Frost
D. C. Rardin
R. G. Staker
R. J. Teunis
C. E. Till
R. S. Zeno
C. E. Dickerman
H. K. Fauske
S. Fistedis
B. D. LaMar
J. F. Marchaterre
H. O. Monson
R. Sevy
G. T. Garvey
J. V. Tokar
M. K. Butler
G. F. Berry
T. R. Bump
S. S. Chen
H. Chung
J. G. Daley
B. J. Hsieh
P. R. Huebotter
J. A. Jendrzejczyk
C. A. Kot (10)
W. P. Lawrence
H. C. Lin
J. J. Lorenz
T. M. Mulcahy
G. S. Rosenberg (5)
Y. S. Shin
Y. W. Shin
M. Srinivasan
P. Turula
R. A. Valentin
T. T. Yeh
C. K. Youngdahl (25)
M. T. Abdel-Moneim
Y. W. Chang
W. L. Chen
G. Nagumo
P. S. Chopra
G. H. Golden
S. Srinivas
A. B. Krisciunas
ANL Contract File
ANL Libraries (5)
TIS Files (6)

External:

ERDA-TIC, for distribution per UC-79h (281)
Manager, Chicago Operations Office
Chief, Chicago Patent Group
Director, Reactor Programs Div., ERDA-CH
Director, ERDA-RDD (2)
Director, CH-INEL
President, Argonne Universities Association
Components Technology Division Review Committee:
- J. W. Dally, Univ. of Maryland
- W. E. Kessler, Commonwealth Associates
- N. C. Rasmussen, Massachusetts Institute of Technology
- M. A. Schultz, Pennsylvania State Univ.
- Alexander Sesonske, Purdue Univ.
- Helmut Thielsch, ITT Grinnell Corp.