4970249 (22)
PD-AAD-889-B1

REPUBLIC OF INDONESIA
MINISTRY OF PUBLIC WORKS
DIRECTORATE GENERAL OF HOUSING BUILDING
PLANNING AND URBAN DEVELOPMENT (CIPTA KARYA)

64p

MEDAN URBAN DEVELOPMENT, HOUSING, WATER SUPPLY AND SANITATION PROJECT

TECHNICAL MEMORANDUM NO.35

DISTRIBUTION SYSTEM ANALYSIS

This is a draft of Appendix E of the Master Plan Report for Water Supply and Sanitation

NOVEMBER 1979

ENGINEERING - SCIENCE, INC. • SINOTECH ENGINEERING CONSULTANTS, INC
A JOINT VENTURE
in association with
PADCO and P.T. DACREA

MEDAN URBAN DEVELOPMENT, HOUSING, WATER SUPPLY AND SANITATION PROJECT
Jln. Singamangaraja 1-3, P.O. Box 28, Phone 20716, Medan-Sumatra, Indonesia

Our ref.: 79/1204/MUDS/313 9 November 1979

Director General Cipta Karya
Ministry of Public Works
Jalan Pattimura 20
Kebayoran Baru
Jakarta Selatan

Subject: Technical Memorandum No. 35, Distribution System Analysis

Dear Sir:

Attached are 20 copies of the subject Technical Memorandum for your information and review. The material contained in this memorandum constitutes submission of Appendix E of the draft Master Plan for Water Supply and Sanitation Report.

Yours faithfully,

ENGINEERING SCIENCE INC. MEDAN URBAN PROJECT INDONESIA

John M. McGill
Project Representative

C.C.: Ir. Ruslan Diwiryo, Director of City & Regional Planning
Ir. Susanto Mertodiningrat, Director of Sanitary Engineering
Ir. Sunaryo, Head Sub-Directorate Town Planning
Ir. K. Pohan, Project Manager, MUDS
AID - Jakarta, attn.: P. Thorn, Project Officer
ES - Arcadia
SINOTECH - Taipei
DACREA - Jakarta

JM/1

BEST COPY AVAILABLE

TABLE OF CONTENTS

MASTER PLAN FOR WATER SUPPLY AND SANITATION

Technical Memorandum No.	Section No.	Section Title
18	1	Introduction
18	2	Existing Water Supply and Sanitation Systems
25	3	Water, Wastewater, Stormwater and Solid Wastes Systems
19	4	Public Health and Environmental Aspects
27	5	Design Criteria and Bases for Cost Estimates
24	6	Alternative Water Supply Management Systems
26	7	Alternative Wastewater Management Systems
23	8	Alternative Drainage Management Systems
20	9	Alternative Solid Wastes Management Systems
28	10	Recommended Water Supply Management System
29	11	Recommended Wastewater Management System
23	12	Recommended Drainage Management System
22	13	Recommended Solid Wastes Management System
30	14	Environmental Assessment
31	15	Finance
*	15	Organization
	Appendix	
34	A	Sibolangit Springs
*	B-1	Surface Water Hydrology
7	B-2	Estimated Stormwater Runoff Curves
7	C-1	Estimated Capacities of Major Rivers
7	C-2	River Channel Cross-sections
32	D-1	Groundwater Resources
8	D-2	Recommended Drilling Program
35	E-1	Network Analysis Computer Program
35	E-2	Computer Program Listing
*	F-1	Demographic Projections
*	F-2	Projected Total Water Demand
*	F-3	Projected Municipal Water Demand
*	F-4	Allocation of Water Demand
*	F-5	Projected Solid Waste Quantities

* Not yet submitted.

LIST OF TECHNICAL MEMORANDA PREVIOUSLY SUBMITTED

Report No.	Title	Date Submitted
1.	Preliminary Site Selection Criteria for Perumnas Medan III, SSCH/LCH Project	March, 1979
2.	Work Plan for Water Supply and Sanitation Master Plans and Feasibility Studies	April, 1979
3.	Policy Issues for Project Guidance	March, 1979
4.	Drainage - Principal Rivers and Rainfall Analysis	May, 1979
5.	Preliminary Evaluation of Sunggal Water Treatment Plant	May, 1979
6.	Public Health Considerations	July, 1979
7.	The Present Drainage System	August, 1979
8.	Recommended Drilling Program	August, 1979
9.	Medan Household Income/Expenditures	August, 1979
10.	Existing Land Use in Medan	August, 1979
11.	The Regional Functions of Medan	August, 1979
12.	Existing Solid Wastes Systems and Projected Solid Wastes Characteristics	August, 1979
13.	Evaluation of Medan Master Plan	August, 1979
14.	Physical Constraints to Urbanization	August, 1979
15.	Development in Health Care in Medan	September, 1979
16.	Existing Water Supply Systems	September, 1979
17.	Developments in Education	October, 1979
18.	Existing Water Supply and Sanitation Systems	October, 1979
19.	Public Health and Environmental Aspects	October, 1979
20.	Alternative Solid Wastes Management Systems	October, 1979

LIST OF TECHNICAL MEMORANDA PREVIOUSLY SUBMITTED (CONTINUED)

Report No.	Title	Date Submitted
21	The Distributed Investment Strategy Plan (DISP)	October, 1979
22	Recommended Solid Wastes Management System	October, 1979
23	Recommended Drainage Management System	October, 1979
24	Alternative Water Supply Management Systems	November, 1979
25	Water, Wastewater, Stormwater and Solid Wastes Systems	November, 1979
26	Alternative Wastewater Management Systems	November, 1979
27	Design Criteria and Bases for Cost Estimates	November, 1979
28	Recommended Water Supply Management System	November, 1979
29	Recommended Wastewater Management System	November, 1979
30	Environmental Assessment	November, 1979
31	Finance	November, 1979
32	Groundwater Resources	November, 1979
33	The Existing Urban Situation	November, 1979
34	Sibolangit Springs	November, 1979

BEST COPY AVAILABLE
BEST COPY AVAILABLE

APPENDIX E

DISTRIBUTION SYSTEM ANALYSIS

E-1 : Program Users Guide

E-2 : Distribution Network Analysis Program Listing

APPENDIX E-1

PROGRAM USERS GUIDE

E-1.1 INTRODUCTION

This program calculates the flow and pressure profile in a hydraulic network. The solution method is based on correcting the flow in each loop until a head loss balance is achieved. Flow corrections are determined from the solution of a set of algebraic equations. The equations are made up from the total differential of the head loss around each elementary loop in the network. The network does not require loop numbering or any special numbering sequence. The program is designed to operate efficiently on large networks: it is presently configured it can accomodate 400 lines and 400 junctions. An option is available to allow adjustment of initial line sizes to fit specific head losses.

E-1.2 MODEL DESCRIPTION

The Hazen Williams flow formula is used as the mathematical model of the hydraulic network. This equation is expressed as described below and utilized in the program to relate pressure loss to flow in a circular conduit. Either USA, USA with imperial gallons, or metric units can be used provided they are consistent and not intermixed. The input data in Section E-1.4 delineates the appropriate units.

$$\Delta P = \frac{(Q)^{1.8518}}{(X)}$$

Where Q = flow in U.S. gpm, imperial gpm, or cubic meters per hour

ΔP = pressure drop between the ends of the pipe in feet or meters

X = pipe characteristic computed from the following formula:

$$X = K \quad C(D)^{2.63} \quad (L)^{-0.54}$$

Where C = Hazen-Williams coefficient of roughness

D = Diameter of pipe in inches or millimeters

L = Length of the pipe in feet or meters

K = Constant

Since the sum of the pressure loss (gain) around each loop in a hydraulic network is zero, equation(1) can be applied to each pipe in each loop to yield a set of non-linear equations that, when solved, will define the pressure profile in the network. That is, each loop will have an equation of the form:

$$H_j = \sum_{k=1}^{\ell} \Delta P_k$$

Where H_j = head loss in loop J

ℓ = number of pipes in loop J

The set of non-linear equations developed by utilizing equation (2) are solved using Newtons methods which is essentially a means of liniarizing a non-linear equation set and thereby permitting the use of one of the many methods available to solve linear simultaneous equations. The procedure is to use the mathematical chain rule to develop the total differential of the loop head loss equation.

$$dH_j = \sum_k \frac{\partial H_j}{\partial q_k} dq_k$$

Where dH_j = total differential of head loss in loop j

dq_k = total differential of flow in pipe k

$\frac{\partial H_j}{\partial (q)_k}$ = partial differential of head loss in loop j with respect to flow in pipe k

Then, assuming $\Delta H_j = dH_j$ and $\Delta q_i = dq_k$ a set of linear equations result that can be solved simultaneously for Δq_i, where Δq_i is the loop flow correction. Using the corrected flow rate a new set of equations are found and another flow correction is determined. This procedure is continued until the loop flow adjustment in all loops is less than some preselected value.

The Choleski method of factorization is used in solving the linear equation set. For symmetric matrices, this method is extremely fast and very accurate.

The mathematical model described above will not, in general, satisfactorily represent an actual hydraulic network. Therefore, pipe fittings and pressure changing elements such as pressure reducing stations (PRV) and booster pumps can be included to obtain a model that is more representative of actual hydraulic systems.

The inclusion of pipe fittings is accomplished by assigning equivalent lengths to the lines. Equivalent lengths are input along with the line data as explained in Section E-1.4.

To include PRV's it is necessary to have both the set point and the pressure loss across the PRV. In this model the network is first solved without PRV's and then the hydraulic grade line (HGL) at the junctions downstream of each of the PRV's is compared to the set point. If the set point is below the calculated HGL then the pressure at the particular junction is maintained at the set point for any additional network solutions. Because controlling of the HGL at a junction may cause unwanted demand flows to occur, this condition is tested for and the model will make adjustments to the PRV set point in an effort to eliminate large demand flows.

The inclusion of booster pumps is accomplished merely by defining the line that contains the pump and the amount of head increase across the pump. These requirements are described in Section E-1.4.

The model assures that reverse flow does not occur in either the PRV's or booster pumps. If reverse flow is found the pressure loss (gain) across the element is adjusted until the reverse flow ceases or the pressure loss (gain) becomes zero. If there is still reverse flow the resistance coefficient for the line that contains the pressure changing element is set to a large value to effectively eliminate flow in that line.

Also included in the model is a means of comparing the calculated head loss in each line to a specified tolerance. If the head loss is not within tolerance, the line diameter is adjusted. After each adjustment the system flows are rebalanced and again head losses are compared to the tolerance. This process is repeated until all head losses are satisfactory or some specified maximum or minimum line diameter is reached.

E-1.3 COMPUTER PROGRAM

The computer program is divided into a MAIN program and 12 subroutines; the subroutines are named INPUT, CHNG, TREE, LOOP, PARAM, NUMBER, MATRIX, ZERO, HYDGL, PROFILE, DIA, and COMP. A functional description of each subroutine is given in the following paragraphs. The algorithms used to define and number loops were devised by R. Epp and A. G. Fowler

Subroutine INPUT. This subroutine reads comments and parameter data and all data required to describe the lines, junctions, and demand flows of the network. Also, a table is established that relates line numbers to junction numbers. Output of line data is also accomplished if a data error is found or if this output option is selected on the parameter card (see Section E-1.4).

Subroutine CHNG. If there is reverse flow in a PRV with zero pressure loss this subroutine removes the set point control from the junction downstream of the PRV. Also, the effective resistance of the line that contains the PRV is set to a large valve; this virtually eliminates the flow in this line.

Subroutine TREE. This subroutine initializes flow in all of the lines. The initialization is such that a mass flow balance at each junction is achieved; however, there may be zero flow in some lines.

Subroutine LOOP. This subroutine identifies all of the network loops.

Subroutine PARAM. This subroutine reads the water source and booster pump data. One card is needed for each water source (there must be at least one in the network) and the first such resource defined becomes the reference head for the network. The water source(s) can be either a pump or reservoir or both. Imaginary loops that connect the reference junction to all constant head junctions are established in this subroutine.

Subroutine NUMBER. This subroutine assigns numbers to the loops in the network. Adjacent loops are numbered sequentially; this tends to increase the number of zero filled diagonals in the coefficient matrix thereby reducing the number of elements that must be considered when solving the equation set.

Subroutine MATRIX. This subroutine calculates the partial differentials of the loop head loss and sets up the coefficient matrix. After calling subroutine COMP to solve the equation set, the flow rates in all lines are adjusted. The amount of flow rate adjustment is compared to the limit set on the parameter input card and if none of the adjustments exceed this limit the solution algorithm is stopped, if the adjustment in one of more lines does exceed the limit a new coefficient matrix is established and another solution is obtained. This iterative proceduring is continued until the flow adjustment test is passed.

Subroutine ZERO. This subroutine is used to test the pressure downstream of all PRV's when they are wide open. If the pressure is above the PRV set point then the pressure at the affected junction is maintained at the set point, if the pressure is below the set point no action is taken.

Subroutine HYDGL. Starting with the given HGL at the reference junction this subroutine calculates the HGL at all the remaining junctions. The procedure is to move from junction to junction in the network while subtracting the head loss in each connecting line.

Subroutine PROFILE. This subroutine generates all of the program output.

Subroutine DIA. This subroutine compares line head loss and makes any necessary adjustments to the line diameter.

Subroutine COMP. This subroutine solves the set of linear simultaneous equations. The results obtained from execution of this subroutine are used to adjust the flows in the network.

E-1.4 PROGRAM INPUT REQUIREMENTS

Preparation of Input Data

1. Prepare a sketch of the network. Show all lines, loops, variable head flow inputs and outputs, booster pumps, PRV's, and water sources.

   A line is a reach of pipe between two junctions.

   A loop is a closed circuit formed by two or more lines.

A junction is a pipe tee, cross, point where water enters or leaves the system, or a point where pipe characteristics change.

A constant head input or output is a point where water enters or leaves the system under a constant pressure while a variable head input or output is a point where flow remains constant.

2. Assign identifying numbers to each line. The sequence of the numbers is of no significance to the program; however, sequential grouping of line numbers in certain regions may be useful to the user when referencing lines.

3. Assign identifying numbers to each junction. The sequence of numbers is not significant to the program so the user can assign the numbers in any convenient manner.

4. Show the flow out(positive) or in (negative) at each junction that does not have a constant head.

5. Show the HGL in all junctions that are to have a constant head.

6. Using the sketch of the network as described above the data can be transcribed to the input forms described below.

Data Requirements

Program input requirements consist of a comment card, a system parameter card, junction data, line data, water source data, and booster pump data. The comment card is the first card in the data set and it can contain a maximum of 80 characters that are used to describe or identify the network being analyzed (see Table E-1.1). The second card in the data stream contains system parameters that are used to establish solution tolerance, adjust demands, limit the number of iterations, control output, indicate whether USA, USA with imperial gallons, or metric units are being used and to specify the head loss tolerance and related parameters.

The solution tolerance (item 1 in Table E-1.2) is used to terminate program execution, i.e. when the flow adjustment in all loops is less than the solution tolerance it is assumed that the network is balanced. If necessary to change all of the demand flows by some amount a factor (item 2 in Table E-1.2) can be entered to accomplish this. For example, if

all flows were to be increased by 25 percent a factor of 1.25 would be entered. The maximum number of iterations (item 3) is used as a means of terminating program execution if the solution procedures is diverging. Items 4 and 5 are self explanatory.

The minimum and maximum head loss allowable (items 6 and 7) is used to determine if a line diameter is too large or too small. The increments (items 8 and 9) are the step changes that will be made to a line diameter if the head loss in that line is not within the allowable. Small and large diameter lines can be varied at different increments if necessary. The diameter that separates small and large is determined by item 10, i.e. all diameters equal to or greater than the value if item 10 are considered large; all other diameters are considered small. There may be some cases in a network where the head loss in a line is fixed by conditions other than line size. For this reason a maximum and minimum diameter allowed can be included in terms 11 and 12 to stop the program from searching for a nonexistent solution.

TABLE E-1.1

COMMENT CARD

Item	Column Position	Format	Unit
Comment	1-80	20A4	NA

TABLE E-1.2

SYSTEM PARAMETER DATA

Item	Column Position		Unit
Solution Tolerance	01-06	F6.0	US, Imperial GPM, or CMH
A factor to adjust the demand flow	07-12	F6.0	NA
Maximum number of iterations	13-16	14 (right justified)	NA
Enter a 0 if the line data is to be printed out, otherwise enter a 1	17-18	12 (right justified)	NA
Enter a 0 for US units, a 1 for metric units, or a 2 for US units with imperial gallons	19-20	12 (right justified)	NA
Minimum head loss allowable	21-26	F6.0	ft/1000 ft or m/1000 m
Maximum head loss allowable	27-32	F6.0	ft/1000 ft or m/1000 m
Increment for small diameter lines	33-38	F6.0	inches or millimeters
Increment for large diameter lines	39-44	F6.0	inches or millimeters
Diameter that separates large and small lines	45-50	F6.0	inches or millimeters
Maximum allowable diameter	51-56	F6.0	inches or millimeters
Minimum allowable diameter	57-62	F6.0	inches or millimeters
Enter a 0 if line diameter adjustments are not allowed; otherwise enter a 1	63-64	12	NA

Junction data follows system parameter data; one card is required for each junction in the network plus a blank card that is used to terminate the junction data input. Table 3 lists the junction data requirements. Item 2 "ADDRESS" can be used to describe the location or any other pertinent information related to a junction.

The PRV requirement is used only for junctions downstream of a PRV. If the PRV is functioning the program will maintain the downstream HGL at this set point. If the PRV is not functioning no action is taken, that is the HGL at the downstream junction is allowed to vary.

Line data follows the junction data; one card is required for each line plus a blank card that is used to terminated the line data. Table E-1.4 described line data input.

TABLE E-1.3

JUNCTION INPUT DATA

Item	Column Position	Format	Unit
Junction Number	01-05	15 (right justified)	NA
Adress	06-25	5A4	NA
Ground Elevation	26-32	F7.0	Feet or Meters
Demand Flow	33-39	F7.0	US GPM, Imperial GPM, CMH
PRV Set Point	40-46	F7.0	Feet or Meters

Following line data is the water source data. One card is needed for each water source (there must be at least one in the network) and the first such source defined becomes the reference head for the network. The water source(s) can be either a pump, a reservoir, or both. Table E-1.5 tabulates the water source data requirements.

Booster pump (or pressure reducing valve) data is next. One card is needed for each pump (or valve) plus a blank card that is used to terminate the booster pump data. See Table E-1.6.

TABLE E-1.4

LINE INPUT DATA

Item	Column Position	Format	Unit
Line Number	01-05	15	NA
Upstream Junction Number	05-10	15	NA
Downstream Junction Number	11-15		
Line length	16-21	F6.0	Feet or Meters
Line Diameter	22-27	F6.0	Inches or Millimeters
Hazen-Williams Coefficient	28-33	F6.0	$(\text{Feet})^{.037}/\text{sec}$
Equivalent Length	34-39	F6.0	Feet or Meters
Enter a 0 for existing line or a 1 for new lines	40-44	15	NA

Sample data sheets are shown as exhibits E-1.I through E-1.IV.

TABLE E-1.5

WATER SOURCE INPUT DATA

Item	Column Position	Format	Unit	Comments
Junction No.	01-05	15 (right justified)	NA	Enter the identifying number of the junction that represents the water source.
Hydraulic Grade Line	11-20	F.10.0	Feet or Metres	Enter the hydraulic grade line of the water source, that is, the elevation of the water surface.
Storage	21-30	F10.0	MG	If a reservoir, then capacity in MG, otherwise 0 for pumpstation.

TABLE E-1.6

BOOSTER PUMP OR PRESSURE REDUCING VALVE INPUT DATA

Item	Column Position	Format	Unit	Comments
Line Number	01-05	I5 (right justified)	NA	Enter the identifying number of the line in which the booster pump installed.
Number of Pumps in Parallel	06-10	I5 (right justified)	NA	Enter the number of pumps in parallel.
Pump Coefficients				Enter the coefficients of a third degree polynominal that expresses the pump head as a function of flow. If a constant head is desired, enter the required head as the first coefficient. Use a negative valve if a pressure reducing valve is used.
	11-20	F10.0	NA	First coefficient
	21-30	F10.0	NA	Second coefficient
	31-40	F10.0	NA	Third coefficient
	41-50	F10.0	NA	Fourth Coefficient

E-1.5. PROGRAM OUTPUT

Program output is shown in the Sample Data Sheets, Exhibits E-1.I to E-1.IV. The first page has title and shows the number of junctions and lines in the network and the total system flow. Also some information is written out to indicate various stages of completion. Vector A in the program is limited to 6,400 elements, hence its length is printed out as an aid to the user.

It was necessary for the program to change the pressure loss (gain) in any of the PRV's or booster pumps, the number of the lines that contain the elements is printed out. The amount of pressure change can be determined by comparing the value printed under the column heading, BOOSTER, or PRV, to the values that were input. Also, if it was necessary to adjust the PRV set point this information is printed out. If adjustments to line diameters were made, the affected line number, the new diameter, and the old unit head loss are printed out. The number of iterations required to balance the network flow is also shown. If this number is one greater than the input iteration limit, the model has not converged to a solution. See Section E-1.6 for a discussion concerning the occurrence of this problem.

The next set of data is junction related, Here the junction number, identification, if any, and the hydraulic grade line, hydraulic pressure, and demand flows are shown.

The next output is the line data. This consists of all information related to each line in the system. A minus sign on the flow and velocity means that the flow direction shown by the from-to node numbers is reversed. For example, in line 2 the flow is from node 157 to node 235 because the flow is positive, whereas in line 1, the flow is from node 160 to node 157 because the flow is negative.

E-1.6 GUIDANCE ON USING THE PROGRAM

The hydraulic network analysis program is easy to use and efficient in terms of computer run time and storage requirements. However, there are some techniques that can be used that will yield even better program performance. These are tabulated below.

1. If there are more than two (2) water sources, choose the reference source in a manner that minimize the distance between the reference and the other sources.

2. Set the solution tolerance (item 1 in Table E-1.3) as loose as possible. Experience with the program indicates a tolerance of 5 to 10 gpm (1.4 to 2.27 cmh) will yield satisfactory answers for most networks. It is advisable, however, to make a test run at, say, 1 gpm and then 5 and then 10 gpm to determine the most appropriate magnitude.

3. The program will normally find a solution in less than 30 iterations; therefore, the iteration limit (item 3 in Table E-1.3) should be set at approximately 30.

There are several data error detecting features in the program to aid the user in identifying data errors. These features are:

1. The junction numbers assigned to the line data are compared to the junction data. If a match is not found, the improperly defined line(s) number is printed out.

2. If a line is isolated from the network, this is noted and printed out.

3. If junctions have been defined and not used, this is printed out including a list of the unused junction numbers.

4. Incorrectly numbered pumps, PRV's, or reservoirs will be noted.

5. If the limit on iterations is reached the solution procedure is probably diverging.

Program Restrictions

1. The maximum number of lines is 400 and the maximum number of junctions is 400.
2. The maximum number of lines that can be connected to a junction is 7
3. The maximum number of booster pumps is 40.
4. The maximum number of reservoirs plus PRV's is 40.
5. The maximum length of vector A is 6,400. The length of vector A is influenced significantly by the number of reservoirs and/or the number of parallel lines.

SAMPLE DATA SHEET
EXHIBIT E-1.1

```
TEST CASE

NO OF JUNCTIONS =    17  NO OF LINES =    17

TOTAL SYSTEM FLOW =    15890.0 GPM OR CMM
CALCULATE INITIAL FLOWS
NETWORK FLOW BALANCE =    0.000
FIND LOOPS
READ RESERVOIR DATA
NUMBER LOOPS
LENGTH OF VECTOR A =    16
SET UP MATRIX

                                        NUMBER OF ITERATIONS=    6

NO OF RESERVOIRS PLUS FIXED HEAD JUNCTIONS =    5

NO OF JUNCTIONS =    17  NO OF LINES =    17

TOTAL SYSTEM FLOW =    0.0 GPM OR CMM
CALCULATE INITIAL FLOWS
NETWORK FLOW BALANCE =    0.002
FIND LOOPS
READ RESERVOIR DATA
NUMBER LOOPS
LENGTH OF VECTOR A =    25
SET UP MATRIX

                                        NUMBER OF ITERATIONS=    8

PRV SET POINT AT JUNCTION    4 HAS BEEN ADJUSTED TO    332.0 BECAUSE DEMAND =    4817.8
                                        NUMBER OF ITERATIONS=    3

PRV SET POINT AT JUNCTION    4 HAS BEEN ADJUSTED TO    334.0 BECAUSE DEMAND =    4420.2
                                        NUMBER OF ITERATIONS=    3

PRV SET POINT AT JUNCTION    4 HAS BEEN ADJUSTED TO    336.0 BECAUSE DEMAND =    3982.3
                                        NUMBER OF ITERATIONS=    3

PRV SET POINT AT JUNCTION    4 HAS BEEN ADJUSTED TO    338.0 BECAUSE DEMAND =    3483.2
                                        NUMBER OF ITERATIONS=    3
```

BEST AVAILABLE DOCUMENT

SAMPLE DATA SHEET
EXHIBIT E-1.II

```
PRV SET POINT AT JUNCTION    4 HAS BEEN ADJUSTED TO      340.0 BECAUSE DEMAND =      2869.7
                                                NUMBER OF ITERATIONS=  8

PRV SET POINT AT JUNCTION    4 HAS BEEN ADJUSTED TO      342.0 BECAUSE DEMAND =      1630.0
                                                NUMBER OF ITERATIONS=  11

PRV SET POINT AT JUNCTION    4 HAS BEEN ADJUSTED TO      344.0 BECAUSE DEMAND =      386.6
                                                NUMBER OF ITERATIONS=  3

LINE    5 ADJUSTED  DIA.=    17. UL =    20.02
                                                NUMBER OF ITERATIONS=  2
```

SAMPLE DATA SHEET
EXHIBIT E-1.III

JUNCTION OUTPUT DATA

JUNCTION NUMBER	ADDRESS	GROUND LEVEL-FT	ELEVATION OF HGL-FT	PRESSURE PSIG	OUTFLOW US GPM
157		250.	370.	52.1	8190.
160		323.	406.	36.0	0.
235		156.	364.	90.1	0.
236		156.	344.	81.4	-252.
294	RES. NO. 2	557.	557.	0.0	-1751.
295		379.	544.	71.5	0.
296		379.	417.	16.5	0.
293	RES. NO. 3	406.	431.	10.8	-12226.
311		264.	510.	106.7	0.
509		250.	460.	90.9	420.
582	EDNA TANK	320.	340.	8.7	671.
820		330.	523.	83.5	970.
850	RES. NO. 1	433.	433.	0.0	-2332.
1047		387.	362.	-10.7	-760.
1045		415.	339.	-32.8	5550.
1056		299.	390.	39.5	0.
1068	TERRANCE ST	395.	339.	-24.2	0.

SAMPLE DATA SHEET
EXHIBIT E-1.IV

LINE DATA OUTPUT

LINE NO.	FROM JUNCTION	TO JUNCTION	H-W COEF.	DIAMETER IN	LENGTH FT	FLOW US GPM	VELOCITY FPS	HEAD LOSS-FT	UNIT LOSS FT/1000FT	BOOSTER HEAD-FT
1	157	160	110.	18.	1910.	-7267.4	-9.16	-35.9	-18.8	0.0
2	157	235	110.	20.	11670.	1409.8	1.44	6.3	0.5	0.0
3	236	582	110.	20.	5460.	1602.0	1.70	4.0	0.7	0.0
4	157	1056	110.	12.	1200.	-2332.5	-6.62	-19.8	-16.5	0.0
5	311	509	110.	17.	3140.	5739.0	8.11	50.4	16.0	0.0
6	509	1047	110.	16.	5220.	5319.0	8.49	97.7	18.7	0.0
7	1045	1047	110.	16.	1640.	-4559.8	-7.27	-23.1	-14.1	0.0
8	582	1068	110.	16.	1000.	991.0	1.58	0.8	0.8	0.0
232	235	236	110.	20.	15.	1409.8	1.44	20.0	1333.9	-20.0
296	293	296	110.	30.	3400.	12225.8	8.55	13.9	4.1	0.0
297	294	295	110.	18.	9600.	1750.5	2.21	12.9	1.3	0.0
298	296	295	110.	30.	20.	4958.4	2.25	-127.0	-6349.2	127.0
299	160	296	110.	27.	4160.	-7267.4	-4.07	-10.9	-2.6	0.0
305	295	820	110.	24.	5300.	6709.8	4.76	21.2	4.0	0.0
306	311	820	110.	24.	4200.	-5739.0	-4.07	-12.6	-3.0	0.0
527	850	1056	110.	14.	5500.	2332.5	4.86	42.9	7.8	0.0
1088	1045	1068	110.	24.	50.	-991.0	-0.70	-0.0	-0.1	0.0

BEST AVAILABLE DOCUMENT

APPENDIX E-2

DISTRIBUTION NETWORK ANALYSIS PROGRAM LISTING

```
C  PROGRAM COMPR COMPARES 2 HYD OUTPUT FILES TO SELECT LARGEST DIAMETER
      DIMENSION COM(20),ADR(5)
       REAL MAXQ
       NHYD=0
      TYPE 401
401   FORMAT(' ENTER OUTPUT DEVICE NAME'/)
      CALL ASSIGN(6,IOUT,-23)
        TYPE 400
400    FORMAT(' ENTER OUTPUT SAVE FILE NAME',/)
      CALL ASSIGN(2,OUT,-23,'NEW',NC,1)
       TYPE 410
410    FORMAT(' ENTER FIRST SAVE FILE NAME'/)
      CALL ASSIGN(3,SAVE,-23,'OLD',NC,1)
      TYPE 420
420   FORMAT(' ENTER SECOND SAVE FILE NAME'/)
      CALL ASSIGN(4,SAV1,-23,'OLD',NC,1)
      READ(3,22) (COM(I),I=1,10)
22    FORMAT(20A4)
      READ(4,22) (COM(I),I=11,20)
       WRITE(2,22) COM
       WRITE(6,22) COM
       READ(3,23) MAXQ,PERCE,MAXIT,IRIT,IMETER,SMIN,SMAX,DINCS,DINCL,
     1DIAT,DMAX,NOPT
      WRITE(2,23) MAXQ,PERCE,MAXIT,IRIT,IMETER,SMIN,SMAX,DINCS,DINCL,
     1DIAT,DMAX,NOPT
      WRITE(6,23) MAXQ,PERCE,MAXIT,IRIT,IMETER,SMIN,SMAX,DINCS,DINCL,
     1DIAT,DMAX,NOPT
       READ(4,23) MAXQ
23    FORMAT(F6.0,F6.4,I4,2I2,6F6.2,I2)
      DO 20 J=1,700
       READ(3,21) NOXY,(ADR(I1),I1=1,5),ELEV,D,TEMP
      WRITE(2,21) NOXY,(ADR(I1),I1=1,5),ELEV,D,TEMP
21     FORMAT(I5,5A4,F7.1,F7.0,F7.1)
      WRITE(6,21) NOXY,(ADR(I1),I1=1,5),ELEV,D,TEMP
       READ(4,21) NO
      IF(NOXY.EQ.0) GO TO 26
20    CONTINUE
26    CONTINUE
      DO 30 I=1,700
       READ(3,31) L1,NO1,NO2,C,D,HW,NEW,ICOST,D3
31    FORMAT(3I5,3F6.0,6X,2I5,F6.0,I5)
       D2=D3
      IF(NHYD.GT.0) D2=D
      D4=D
      D5=D3
      IF(NHYD.GT.1.AND.ICOST.EQ.0) D2=0.
      J=3
      HW2=HW
      ICOST1=ICOST
       READ(4,31) L1,NO1,NO2,C,D,HW,NEW,ICOST,D3
      D6=D3
```

BEST AVAILABLE DOCUMENT

```
      IF(NHYD.GT.0) D6=D
       IF(NHYD.LT.1.AND.ICOST.EQ.0) GO TO 35
      IF(D6.LT.D2) GO TO 35
      D4=D
      D5=D3
      J=4
      Hw2=HW
      ICOST1=ICOST
35    WRITE(2,31) LI,NO1,NO2,C,D4,Hw2,NEW,ICOST1,D5
      WRITE(6,31) LI,NO1,NO2,C,D4,Hw2,NEW,ICOST1,D5,J
      IF(LI.EQ.0) GO TO 40
  30   CONTINUE
40    CONTINUE
      DO 100 I=1,40
      READ(3,102) K,L,COR4
      WRITE(2,102) K,L,COR4
      WRITE(6,102) K,L,COR4
      IF(K.EQ.0) GO TO 101
100    CONTINUE
101   CONTINUE
102   FORMAT(2I5,5F10.0)
       END
```

BEST AVAILABLE DOCUMENT

```
      SUBROUTINE CHNG
      COMMON COR4S(20),SMIN,SMAX,DINCS,DINCL,COR4(20),DELP(200),
     2COB1(20),COB2(20),COB3(20),COB0(20),A(6400),MAXQ,D,
     3RP,HEAD,RP1,RP2,RP3,RHD,MET1,DIAT,DMAX,
     4NOLAL(1200),J,K,JJ,IOUT,IIN,NRES,NOJUC(20),NRV(20),NRV1(20),
     5NIL(20),NOSD1(402),NLNJ(401),LNJ(800),
     6NOJT(20),NRN(20),N,LINES,JUNCS,NO1,NO2,NOPT,
     7IDEX,NORS,NTM,ITM1,ITM2,LPKT,NL,ITJ1,ITJ2,KKK,NOBP,
     8NR,NODT,L1,L2,L3,KK,LP,LNTH,NOLJ1,IRIT,IMETER,IOUT2,
     9MAXIT,IC,IC1,IFIRST,NOPRV,ICIS
       EQUIVALENCE(NOLJ(1),NOSD(2),NOSD1(3)),(NOPIP(1),LPRP(1),NOLAL(1))
     1,(NOLAL(401),NLJJ(1))
       EQUIVALENCE (A(1),QTJU(1),IP(1)),(A(401),NLT(1),IDL(1)),
     1(A(801),IL1(1),IPL(1)),(A(1201),LLIN(1),NNST(1)),
     2(A(1601),JUNCL(1)),(A(2001),IL(1)),(A(2401),HDGL(1),NX(1))
     3,(A(2801),ILDK(1)),(A(4001),NUM(1)),(A(5201),ILOP(1))
     4,(A(5401),NOST(1)),(A(5801),NOST1(1))
       DIMENSION NOLJ(400),NOSD(401),QTJU(400),HDGL(400),NOST1(200),
     1IP(400),IDL(400),NLT(400),IL1(400),IPL(400),LLIN(400),NNST(400),
     2NLJJ(400),JUNCL(400),IL(400),NX(400),LPRP(400),ISAVE(20),
     3NOPIP(400),ILDK(1200),NUM(1200),ILOP(200),NOST(400)
       REAL MAXQ,MET1
C IF COB0 IS ZERO AND THER IS REVERSE FLOW
C SET COEF=1 AND REMOVE SET POINTS
       J1=0
       DO 22 J=1,NORS
       IJ=NOJUC(J)
       DO 23 I=ICIS,IC1
       LP=NRV1(I)
      READ(8'LP) L1,NO2,NO1,COMPR,D1,DO,C,T1,COEF,Q,NEW
       IF(NO1.EQ.IJ) GO TO 10
23     CONTINUE
       J1=J1+1
       NOJUC(J1)=IJ
       COR4(J1)=COR4(J)
       GO TO 22
10    COEF=1.
      WRITE(8'LP) L1,NO1,NO2,COMPR,D1,DO,C,T1,COEF,Q,NEW
22     CONTINUE
       J2=NORS-J1
       NRES=NRES-J2
       NORS=NRES-1
       RC=SUM/FLOAT(NRES)
      READ(4'NR) NOXYT,(A(I1),I1=1,5),EL,DEMAND
      DEMAND=-RC
      WRITE(4'NR) NOXYT,(A(I1),I1=1,5),EL,DEMAND
      DO 24 I=1,NORS
       L=NOJUC(I)
      READ(4'L) NOXYT,(A(I1),I1=1,5),EL,DEMAND
      DEMAND=-RC
      WRITE(4'L) NOXYT,(A(I1),I1=1,5),EL,DEMAND
24    CONTINUE
       DO 26 I=ICIS,IC1
```

```
      LP=NRV1(1)
      READ(8'LP) LT,NO2,NO1
      READ(4'NO1) NOXYT,(A(I1),I1=1,5),EL,DEMAND
      DEMAND=0.
      WRITE(4'NO1) NOXYT,(A(I1),I1=1,5),EL,DEMAND
26    CONTINUE
      IC1S=IC1+1
      RETURN
      END
```

MICROFILMED FROM BEST
AVAILABLE COPY

MICROFILMED FROM BEST
AVAILABLE COPY

```
      SUBROUTINE COMPAR
      COMMON COR45(20),SMIN,SMAX,DINCS,DINCL,COR4(20),DELP(200),
     2COB1(20),COB2(20),COB3(20),COB0(20),A(6400),MAXQ,D,
     3RP,HEAD,RP1,RP2,RP3,RHO,MET1,DIAT,DMAX,
     4NOLAL(1200),J,K,JJ,IOUT,IIN,NRES,NOJUC(20),NRV(20),NRV1(20),
     5MIL(20),NOSD1(402),NLNJ(401),LNJ(800),
     6NOJT(20),NRN(20),N,LINES,JUNCS,NO1,NO2,NOPT,
     7IDEX,NORS,ITM,ITM1,ITM2,LPST,ML,ITJ1,ITJ2,KKK,NOBP,
     8NR,NODT,L1,L2,L3,KK,LP,LNTH,NOLJ1,IRIT,IMETER,IOUT2,
     9MAXIT,IC,IC1,IFIRST,NOPRV,IC1S
      EQUIVALENCE(NOLJ(1),NOSD(2),NOSD1(3)),(NOPIP(1),LPRP(1),NOLAL(1))
     1,(NOLAL(401),NLJJ(1))
      EQUIVALENCE (A(1),QTJU(1),IP(1)),(A(401),NLT(1),IDL(1)),
     1(A(801),ILI(1),IPL(1)),(A(1201),LLIN(1),NNST(1)),
     2(A(1601),JUNCL(1)),(A(2001),IL(1)),(A(2401),HDGL(1),NX(1))
     3,(A(2801),INDK(1)),(A(4001),NUM(1)),(A(5201),ILOP(1))
     4,(A(5401),NOST(1)),(A(5801),NOST1(1))
      DIMENSION NOLJ(400),NOSD(401),QTJU(400),HDGL(400),NOST1(200),
     1IP(400),IDL(400),NLT(400),ILI(400),IPL(400),LLIN(400),NNST(400),
     2NLJJ(400),JUNCL(400),IL(400),NX(400),LPRP(400),ISAVE(20),
     3NOPIP(400),INDK(1200),NUM(1200),ILOP(200),NOST(400)
      REAL MAXQ,COM(20),MET1
      PERCE=1.
      WRITE(9,22) COM
22    FORMAT(20A4)
      WRITE(9,23) MAXQ,PERCE,MAXIT,IRIT,IMETER,SMIN,SMAX,DINCS,
     1DINCL,DIAT,DMAX,NOPT
23    FORMAT(2F6.1,I4,2I2,6F6.2,I2)
      DO 20 J=1,JUNCS
       READ(4'J) NOXYT,(A(I1),I1=1,5),RC,D
       DO 40 J1=1,NRES
       IF(NRN(J1).EQ.J) D=-1.
40     CONTINUE
20     WRITE(9,21) NOXYT,(A(I1),I1=1,5),RC,D,TEMP
21      FORMAT(I5,5A4,F7.1,F7.0,F7.1)
        I=0
       WRITE(9,21) I
31     FORMAT(3I5,3F6.0,6X,2I5,F6.0)
       WRITE(IOUT,100)
100    FORMAT(1H1,'LINE NO  EXISTING  NEW    EQUIVALENT')
       DO 10 I=1,LINES
       READ(8'I) DT,NO1,NO2,COMPR,D1,DO,HW,T1,COEF,Q,NEW
       D=D1
       ICOST=1
       IF(NEW.GT.0) GO TO 11
       ICOST=0
       IF(DO.EQ.0) GO TO 11
       PERCE=COEF*ABS(Q)**1.8518
       Q1=HW*DO**2.63*(PERCE/(10.41*MET1*COMPR))**.54
       Q2=ABS(Q)-Q1
       D=(Q2/ABS(Q))**.3602*D1
       ID=D/10.+.5
```

```
      D=ID*10
      WRITE(TOUT,110) LT,DO,D1,D
110   FORMAT(1H ,I5,3F10.1)
      ICOST=1
11    CONTINUE
      READ(4'NO1) NOXYT
      READ(4'NO2) N2
      WRITE(9,31) LT,NOXYT,N2,COMPR,D1,HW,NEW,ICOST,D
10    CONTINUE
      I=0
      WRITE(9,21) I
      RETURN
      END
```

BEST COPY AVAILABLE

BEST COPY AVAILABLE

```
      SUBROUTINE ZERO
      COMMON COR4S(20),SMIN,SMAX,DINCS,DINCL,COR4(20),DELP(200),
     2CDB1(20),CDB2(20),CDB3(20),CDB0(20),A(6400),MAXQ,D,
     3RP,HEAD,RP1,RP2,RP3,RHD,MET1,DIAT,DMAX,
     4NOLAL(1200),J,K,JJ,IOUT,IIN,NRES,NOJUC(20),NRV(20),NRV1(20),
     5NIL(20),NOSD1(402),NLNJ(401),LNJ(800),
     6NOJT(20),NRN(20),N,LINES,JUNCS,NO1,NO2,NOPT,
     7IDEX,NORS,NTM,ITM1,ITM2,LPKT,NL,ITJ1,ITJ2,KKK,NOBP,
     8NR,NODT,L1,L2,L3,KK,LP,LNTH,NOLJ1,IRIT,IMETER,IOUT2,
     9MAXIT,IC,IC1,IFIRST,NOPRV,IC1S
      EQUIVALENCE(NOLJ(1),NOSD(2),NOSD1(3)),(NOPIP(1),LPRP(1),NOLAL(1))
     1,(NOLAL(401),NLJJ(1))
      EQUIVALENCE (A(1),QTJU(1),IP(1)),(A(401),NLT(1),IDL(1)),
     1(A(801),ILI(1),IPL(1)),(A(1201),LLIN(1),NNST(1)),
     2(A(1601),JUNCL(1)),(A(2001),IL(1)),(A(2401),HDGL(1),NX(1))
     3,(A(2801),INDK(1)),(A(4001),NUM(1)),(A(5201),ILOP(1))
     4,(A(5401),NOST(1)),(A(5801),NOST1(1))
      DIMENSION NOLJ(400),NOSD(401),QTJU(400),HDGL(400),NOST1(200),
     1IP(400),IDL(400),NLT(400),ILI(400),IPL(400),LLIN(400),NNST(400),
     2NLJJ(400),JUNCL(400),IL(400),NX(400),LPRP(400),ISAVE(20),
     3NOPIP(400),INDK(1200),NUM(1200),ILOP(200),NOST(400)
      REAL MAXQ,MET1
      IC1=0
C IF DOWN STREAM HGL IS LOWER THAN UPSTR
C  SET PRV SET POINT
      DO 10 J=1,NOBP
      LP=NRN(J)
      IF(CDB0(J).GT.0.) GO TO 10
     READ(8'LP) LT,NO2,NO1
      DO 26 JJ=1,NOPRV
      K=NOJT(JJ)
      IF(K.EQ.NO1) GO TO 27
26    CONTINUE
27    K=JJ
      IF(HDGL(NO1).LT.COR4S(K)) GO TO 22
      IC1=1
      NORS=NORS+1
      COR4(NORS)=COR4S(K)
      NOJUC(NORS)=NO1
      IFIRST=2
      GO TO 10
22    CONTINUE
      WRITE(IOUT,31) NO1,HDGL(NO1)
31    FORMAT(1H ,'HGL AT ',I5,' IS',F10.1)
      CDB0(J)=0.
10    CONTINUE
      NRES=NORS+1
      WRITE(IOUT,50) NRES
50    FORMAT(1H ,'NO OF RESERVOIRS PLUS FIXED HEAD JUNCTIONS = ',I5)
      RETURN
      END
```

BEST COPY AVAILABLE

```
      SUBROUTINE DIA(IDIA,IDIAC)
      COMMON CORAS(20),SMIN,SMAX,DIACS,DINCL,CORA(20),DELP(200),
     2CORB1(20),CORB2(20),CORB3(20),CORB0(20),A(6400),MAXQ,D,
     3RP,HEAD,RP1,RP2,RP3,RHO,MET1,DIAT,DMAX,
     4NOLAL(1200),J,K,JJ,IOUT,IIN,NRES,NOJUC(20),NRV(20),NRV1(20),
     5NIL(20),NOSD1(402),NLNJ(401),LNJ(800),
     6NOJT(20),NFR(20),N,LINES,JUNCS,NO1,NO2,NOPT,
     7IDEX,NORS,NIM,ITM1,ITM2,LPRT,DL,ITJ1,ITJ2,KKK,NOBP,
     8NR,ICD1,L1,L2,L3,KK,LP,LNTH,NOLJ1,IKIT,IMETER,IOUT2,
     9MAXIT,IC,IC1,IFIRST,NOPRV,IC1S
      EQUIVALENCE(NOLJ(1),NOSD(2),NOSD1(3)),(NOPIP(1),LPRP(1),NOLAL(1))
     1,(NOLAL(401),NLJJ(1))
      EQUIVALENCE (A(1),QTJU(1),IP(1)),(A(401),NLT(1),IDL(1)),
     1(A(801),TL1(1),IPL(1)),(A(1201),LLIN(1),NNST(1)),
     2(A(1601),JUNCL(1)),(A(2001),IL(1)),(A(2401),HDGL(1),NX(1))
     3,(A(2801),INDK(1)),(A(4001),NUM(1)),(A(5201),ILOP(1))
     4,(A(5401),NUST(1)),(A(5801),NUST1(1))
      DIMENSION NOLJ(400),NOSD(401),QTJU(400),HDGL(400),NOST1(200),
     1IP(400),IDL(400),NLT(400),TL1(400),IPL(400),LLIN(400),NNST(400),
     2NLJJ(400),JUNCL(400),IL(400),NX(400),LPRP(400),ISAVE(20),
     3NOPIP(400),INDK(1200),NUM(1200),ILOP(200),NUST(400)
      REAL MAXQ,COM(20),MET1
      DMIN=20.
      EOL=0.
      IDIA=1
      IDIAC=IDIAC+1
      IF(IDIAC.GT.15) RETURN
      DO 150 I=1,LINES
      II=I
      READ(8'II) LT,NO1,NO2,COMPR,D1,DO,C,T1,RC,Q,NEW
      TEMP=10.41*MET1*(COMPR+EOL)
      IF(D1.GT.DMAX) GO TO 150
      PERCE=RC*Q*ABS(Q)**.8518
      DL=ABS(PERCE)/COMPR*1000.
      IF(DL.LT.SMIN.OR.DL.GT.SMAX) GO TO 160
      GO TO 150
160   ADD=DINCL
200   FORMAT(' LINE ',I5,' ADJUSTED  DIA.= ',F6.0,' DL = ',F8.2)
      IF(D1.LT.DIAT) ADD=DIACS
      IF(DL.LT.SMIN) GO TO 170
      D1=D1+ADD
      IDIA=0
      WRITE(IOUT,200) LT,D1,DL
      RC=TEMP/(C*D1**2.63)**1.8518
      WRITE(8'II)LT,NO1,NO2,COMPR,D1,DO,C,T1,RC,Q,NEW
      GO TO 150
170   IF(NEW.EQ.0) GO TO 150
      D1=D1-ADD
      IF(D1.LT.DMIN) GO TO 180
      IDIA=0
      GO TO 181
180   D1=20.
```

BEST COPY AVAILABLE

```
181     WRITE(TOUT,200) LT,D1,DL
        RC=TEMP/(C*D1**2.63)**1.8518
        WRITE(8'11) LT,ND1,ND2,COMPR,D1,DD,C,T1,RC,Q,NEW
150     CONTINUE
        RETURN
        END
```

BEST COPY AVAILABLE

```
      SUBROUTINE MATRIX
      COMMON COR4S(20),SMIN,SMAX,DINCS,DINCL,COR4(20),DELP(200),
     2COB1(20),COB2(20),COB3(20),COB0(20),A(6400),MAXQ,D,
     3RP,HEAD,RP1,RP2,RP3,RHD,MET1,DIAT,DMAX,
     4NOLAL(1200),J,K,JJ,IOUT,IIN,NRES,NOJUC(20),NRV(20),NRV1(20),
     5NIL(20),NOSD1(402),NLNJ(401),LNJ(800),
     6NOJT(20),NRN(20),N,LINES,JUNCS,NO1,NO2,NOPT,
     7IDEX,NORS,NTM,ITM1,ITM2,LPKT,NL,ITJ1,ITJ2,KKK,NOBP,
     8NR,NOD1,L1,L2,L3,KK,LP,LNTH,NOLJ1,IRIT,IMETER,IOUT2,
     9MAXIT,IC,IC1,IFIRST,NOPRV,IC1S
       EQUIVALENCE(NOLJ(1),NOSD(2),NOSD1(3)),(NOPIP(1),LPRP(1),NOLAL(1))
     1,(NOLAL(401),NLJJ(1))
       EQUIVALENCE (A(1),QTJU(1),IP(1)),(A(401),NLT(1),IDL(1)),
     1(A(801),ILI(1),IPL(1)),(A(1201),LLIN(1),NNST(1)),
     2(A(1601),JUNCL(1)),(A(2001),IL(1)),(A(2401),HDGL(1),NX(1))
     3,(A(2801),INDK(1)),(A(4001),NUM(1)),(A(5201),ILOP(1))
     4,(A(5401),NOST(1)),(A(5801),NOST1(1))
       DIMENSION NOLJ(400),NOSD(401),QTJU(400),HDGL(400),NOST1(200),
     1IP(400),IDL(400),NLT(400),ILI(400),IPL(400),LLIN(400),NNST(400),
     2NLJJ(400),JUNCL(400),IL(400),NX(400),LPRP(400),ISAVE(20),
     3NOPIP(400),INDK(1200),NUM(1200),ILOP(200),NOST(400)
       REAL MAXQ,C(5),MET1
       CNTR=1./1.8518
       WRITE(IOUT,1)
 1     FORMAT(1H ,'SET UP MATRIX')
1825   ICOUT=0
       ICTEST=0
1827   DO 42 I=1,20
       NRV1(I)=0
42     NRV(I)=0
       ITER=0
1826   DO 111 I=1,LNTH
111    A(I)=0.0
       DO 114 I=1,N
 114   DELP(I)=0.
       DO 116 I=1,LINES
       L1=NOSD1(I)
       L2=NOSD1(I+1)-1
       IF(L1.GT.L2) GO TO 116
      READ(K'1) L1,NO1,NO2,COMPR,D1,D0,HW,T1,COEF,PRO,NEW
       RC=COEF*ABS(PRO)**.8518
       BMQ=RC*PRO
       GO TO 1410
130    IF(L1.GT.L2) GO TO 116
1410   J=NOLAL(L1)
       JJ=IABS(J)
       S=1.0
       IF(J.GT.0) S=-1.0
       DELP(JJ)=DELP(JJ)+BMQ*S
       KKK=(JJ-1)*NOD1+1
       DO 131 L=L1,L2
       K=NOLAL(L)
```

BEST COPY AVAILABLE

```
      KK=IABS(K)
     IDEX=KKK+KK-JJ
      S=1.0
      IF(J*K.LT.0) S=-1.0
      A(IDEX)=A(IDEX)+S*RC
131   CONTINUE
      L1=L1+1
      GO TO 130
116   CONTINUE
     IF(NORS.EQ.0) GO TO 4021
      D=D/RP
      ABD=ABS(D)
      T=HEAD*(1.0+D*(RP1+ABD*(RP2+ABD*RP3)))+RHD
      D=D*RP
      DO 1703 I=1,NORS
      K=NIL(I)
      L=NOJUC(I)
     READ(4'L) NOXYT,(C(II),II=1,5),EL,DEMAND
     DND=DEMAND
      ABD=ABS(DND)
     DELP(K)=DELP(K)+T-COR4(I)
      ITS=(K-1)*NODI +1
     A(ITS)=A(ITS)+RP1*HEAD
1703  CONTINUE
4021 CONTINUE
      IF(NOBP.EQ.0) GO TO 1716
      DO 1717 I=1,NOBP
      T=0.
      RC=0.
      IF(COB0(I).LT.0..AND.IFTRST.EQ.1) GO TO 1720
      LP=NRN(I)
     READ(8'LP) LT,NO1,NO2,COMPR,D1,DO,HW,T1,COEF,Q,NEW
      PRQ=Q
      ABQ=ABS(PRQ)
      T=-(COB0(I)*(1.-PRQ*(COB1(I)+ABQ*(COB2(I)+ABQ*(COB3(I))))))
     T1=-T
     WRITE(8'LP) LT,NO1,NO2,COMPR,D1,DO,HW,T1,COEF,Q,NEW
      RC=COB0(I)*(COB1(I)+ABQ*(2.*COB2(I)+ABQ*(3.*COB3(I))))
      RC=RC*CNTR
1720  L1=NOSD1(LP)
      L2=NOSD1(LP+1)-1
1719  IF(L1.GT.L2) GO TO 1717
      J=NOLAL(L1)
      JJ=IABS(J)
      S=1.0
      IF(J.GT.0) S=-1.0
      DELP(JJ)=DELP(JJ)+S*T
      KKK=(JJ-1)*NODI+1
      DO 1718 L=L1,L2
      K=NOLAL(L)
      KK=IABS(K)
      S=1.0
```

BEST COPY AVAILABLE

```
      IF(GI*K.LT.0) S=-1.0
      IDEX=KKK+KK-JJ
      A(IDEX)=A(IDEX)+S*RC
1718  CONTINUE
      L1=L1+1
      GO TO 1719
1717  CONTINUE
1716  CONTINUE
      DO 1704 I=1,N
      DELP(I)=DELP(I)*CNTR
1704  CONTINUE
      CALL COMP(A,DELP,N,NODI,IOUT)
      DOM=0.0
      DO 1707 I=1,N
      DQ=ABS(DELP(I))
      IF(DQ.GT.DOM) DOM=DQ
1707  CONTINUE
      DO 1705 I=1,LINES
      L1=NOSD1(I)
      L2=NOSD1(I+1)-1
      IF(L1.GT.L2) GO TO 1705
      T=0.0
      DO 1706 L=L1,L2
      J=NOLAL(L)
      S=1.0
      IF(J.LT.0) S=-1.0
      J=IABS(J)
      T=T+S*DELP(J)
1706  CONTINUE
      READ(8'I) LT,NO1,NO2,COMPR,D1,DO,HW,T1,COEF,O,NEW
      O=O+T
      WRITE(8'I) LT,NO1,NO2,COMPR,D1,DO,HW,T1,COEF,O,NEW
1705  CONTINUE
      IF(NOES.EQ.0) GO TO 1708
      DO 1709 I=1,NOES
      K=NOLDUC(I)
      J=MIL(I)
      DQ=DELP(J)
      READ(4'K) NOXYT,(C(II),II=1,5),EL,DEMAND
      DEMAND=DEMAND+DQ
      WRITE(4'K)NOXYT,(C(II),II=1,5),EL,DEMAND
      D=D-DQ
1709  CONTINUE
      READ(4'NR) NOXYT,(C(II),II=1,5),EL,DEMAND
      WRITE(4'NR) NOXYT,(C(II),II=1,5),EL,D
1708  CONTINUE
      ITER=ITER+1
      IF(ITER.GT.MAXIT) GO TO 2
      IF(DOM.GT.MAXQ) GO TO 1826
2     WRITE(IOUT,1984) ITER
1984  FORMAT(1H0,50X,'NUMBER OF ITERATIONS=',I5//)
      IF(NOSP.EQ.0) GO TO 3815
      IF(NOPRV.EQ.0) GO TO 1828
```

BEST COPY AVAILABLE

```
      IF(IFIRST.NE.1) GO TO 1828
      RETURN
1828    IFIRST=3
      IC=0
      IC1=0
C FIND LINES WITH REVERSE FLOW
      DO 2000 J=1,NOBP
      LP=NRN(J)
      IF(COBO(J).NE.0.) GO TO 2101
      IC1=IC1+1
      NRVT(IC1)=LP
      GO TO 2000
2101   READ(8'LP) LT,NO1,NO2,COMPR,D1,DO,HW,T1,COEF,Q,NEW
      IF(Q.GT.0.) GO TO 2000
      IC=IC+1
      NRV(J)=LP
2000  CONTINUE
      IF(IC.EQ.0) GO TO 2002
      ICTEST =ICTEST+1
      IF(ICTEST.GT.20) GO TO 4815
       WRITE(IOUT,29)
  29  FORMAT(1H ,'PRESSURE DROP (LOSS) HAS BEEN ADJUSTED IN THE FOLLOWI
     1NG LINES')
C ADJUST PRESS. DROP IN REVERSE FLOW LINES
      DO 28 I=1,NOBP
      L=NRV(I)
       IF(L.EQ.0) GO TO 28
      LP=L
      COBO(I)=COBO(I)+1.
      IF(ABS(COBO(I)).LT.2.) COBO(I)=0.
      WRITE(IOUT,27) LP
27    FORMAT(1H ,I5)
 28   CONTINUE
      GO TO 1827
2002  CONTINUE
C RETURN IF THERE IS REVERSE FLOW AND COBO HAS BEEN REDUCED TO ZERO
C SO COEF CAN BE SET = 1 USING SUBROUTINE CHNG
      IF(IC1.GE.IC1S) RETURN
      ICOUT=ICOUT+1
      IF(ICOUT.GT.8) GO TO 4815
      ICOUNT=0
      IF(NOPRV.EQ.0) GO TO 3815
      DO 30 IJ=1,NOPRV
      NO1=NOJT(IJ)
      DO 50 J=1,NORS
      NO2=NOJUC(J)
      IF(NO1.EQ.NO2) GO TO 51
50    CONTINUE
      GO TO 30
51    K=J
C ADJUST PRV SET POINTS TO REDUCE FLOW
      READ(4'NO1) NOXYT,(C(I1),I1=1,5),EL,DEMAND
```

BEST COPY AVAILABLE

```
      IF(DEMAND.GT.300.) GO TO 32
      GO TO 31
32     COR4(K)=COR4(K)+2.
      WRITE(IOUT,7000) NO1,COR4(K),DEMAND
      ICOUNT=1
31    CONTINUE
      READ(4'NO1) FOXYT,(C(I1),I1=1,5),EL,DEMAND
      IF(DEMAND.LT.-300.) GO TO 33
      GO TO 30
33     COR4(K)=COR4(K)-2.
      WRITE(IOUT,7000) NO1,COR4(K),DEMAND
7000  FORMAT(1H ,'PRV SET POINT AT JUNCTION ',I5,' HAS BEEN ADJUSTED TO'
     1,F12.1,' BECAUSE DEMAND = ',F12.1)
      ICOUNT=1
30    CONTINUE
      IF(ICOUNT.EQ.1) GO TO 1827
      GO TO 3815
 4815   WRITE(IOUT,4816)
4816  FORMAT(1H ,'THER IS EITHER NEGATIVE FLOW OR TOO HIGH DEMAND AT A
     1FIXED HEAD JUNCTION')
3815  IF(NOPT.EQ.0) GO TO 3816
      CALL DIA(IDIA,IDTAC)
      IF(IDIA.EQ.0) GO TO 1825
3816  TC1=0
      RETURN
      END
```

BEST COPY AVAILABLE

```
      SUBROUTINE PROFIL
      COMMON COR45(20),SMIN,SMAX,DINCS,DINCL,COR4(20),DELP(200),
     2COB1(20),COB2(20),COB3(20),COB0(20),A(6400),MAXQ,D,
     3RP,HEAD,RP1,RP2,RP3,RHO,MET1,DIAT,DMAX,
     4NOLAL(1200),J,K,JJ,IOUT,IIN,NREF,NOJUC(20),NRV(20),NRV1(20),
     5NIL(20),NOSD1(402),NLNJ(401),LNJ(800),
     6NOJT(20),NRH(20),N,LINES,JUNCS,NO1,NO2,NOPT,
     7IDEX,NORS,NTM,ITM1,ITM2,LPKT,NL,ITJ1,ITJ2,KKK,NOBP,
     8NR,NODI,L1,L2,L3,KK,LP,LNTH,NOLJ1,IRIT,IMETER,IOUT2,
     9MAXIT,IC,IC1,IFIRST,NOPRV,IC1S
      EQUIVALENCE(NOLJ(1),NOSD(2),NOSD1(3)),(NOPIP(1),LPRP(1),NOLAL(1))
     1,(NOLAL(401),NLJJ(1))
      EQUIVALENCE (A(1),QTJU(1),IP(1)),(A(401),NLT(1),IDL(1)),
     1(A(801),ILI(1),IPL(1)),(A(1201),LLIN(1),NNST(1)),
     2(A(1601),JUNCL(1)),(A(2001),IL(1)),(A(2401),HDGL(1),NX(1))
     3,(A(2801),INDK(1)),(A(4001),NUM(1)),(A(5201),ILOP(1))
     4,(A(5401),NOST(1)),(A(5801),NOST1(1))
      DIMENSION NOLJ(400),NOSD(401),QTJU(400),HDGL(400),NOST1(200),
     1IP(400),IDL(400),NLT(400),ILI(400),IPL(400),LLIN(400),NNST(400),
     2NLJJ(400),JUNCL(400),IL(400),NX(400),LPRP(400),ISAVE(20),
     3NOPIP(400),INDK(1200),NUM(1200),ILOP(200),NOST(400)
      DATA FL/' US ','IMP '/
      READ MAXQ,MET1,FL(2)
      TEMP=3.5368
      TEMP1=1.
      IF(IMETER.EQ.1) GO TO 3
      TEMP=.4085
      TEMP1=.433
      IK=1
      IF(IMETER.EQ.0) GO TO 3
      TEMP=.49059
      TEMP1=.433
      IK=2
3     CONTINUE
      I=0
      KK=(JUNCS-1)/22+1
      DO 2 KKK=1,KK
      IDEX=JUNCS-I
      IF(IDEX.GT.22) IDEX=22
      WRITE(IOUT,1001)
1001  FORMAT(101,////,37X,'JUNCTION OUTPUT DATA')
      WRITE(IOUT,1002)
1002  FORMAT(140,'JUNCTION',T17,'ADDRESS',T33,'GROUND    ELEVATION
     1  PRESSURE      OUTFLOW')
      IF(IMETER.EQ.1) GO TO 7
6     WRITE(IOUT,8) FL(IK)
8     FORMAT(140,T3,'NUMBER',T31,' LEVEL-FT   OF HGL-FT',9X,'PSIG',
     17X,A4,'GPM')
      GO TO 9
7     WRITE(IOUT,1003)
1003  FORMAT(140,T3,'NUMBER',T31,' LEVEL-M    OF HGL-M ',8X,'METERS',
     16X,' CMH    ')
```

BEST COPY AVAILABLE

```
9       CONTINUE
        DO 146 JJ=1,IDEX
        I=I+1
       READ(4'I) NX1,(CQB1(I1),I1=1,5),RC,DEMAND
       NLNJ(I)=NX1
        PRESH=(HDGL(I)-RC)*TEMP1
        WRITE(IOUT,900) NX1,(CQB1(I1),I1=1,5),RC,HDGL(I),PRESH,DEMAND
146     CONTINUE
 900    FORMAT(1H0,T1,I5,T11,5A4,T33,F6.0, 5X,F6.0,9X,F7.1,6X,F8.0)
 2      CONTINUE
        I=0
        KK=(LINES-1)/22+1
        DO 1 KKK=1,KK
        IDEX=LINES-I
        IF(IDEX.GT.22) IDEX=22
        WRITE(IOUT,910)
  910   FORMAT(1H1,////,49X,'LINE DATA OUTPUT')
        WRITE(IOUT,911)
 911    FORMAT(1H0,3X,'LINE',6X,'FROM',9X,'TO',9X,'H-W',5X,'DIAMETER',  (
       (3X,'LENGTH',5X,'FLOW',5X,'VELOCITY',5X,'HEAD',5X,'UNIT LOSS',
       23X,'BOOSTER')
        IF(IMETER.EQ.1) GO TO 4
       WRITE(IOUT,5) FL(IK)
        GO TO 10
 5      FORMAT(1H0,4X,'NO.',4X,'JUNCTION',4X,'JUNCTION',5X,'COEF.',
       17X,'IN',8X,'FT',5X,A4,'GPM',7X,'FPS',6X,'LOSS-FT',3X,'FT/1000FT',
       3'    HEAD-FT')
 4      WRITE(IOUT,912)
 912    FORMAT(1H0,4X,'NO.',4X,'JUNCTION',4X,'JUNCTION',5X,'COEF.',
       17X,'CM',6X,'METERS',3X,'  CMH  ' ,5X,'MPS',7X,'LOSS-M',5X,'M/1000M
       1     HEAD-M')
 10     CONTINUE
        DO 150 JJ=1,IDEX
        I=I+1
       READ(8'I) LT,NO1,NO2,COMPR,D1,DD,C,T1,RC,PRO,NEW
        IF(NO1.EQ.0) GO TO 150
        NX1=NLNJ(NO1)
        NX2=NLNJ(NO2)
       V=PRO*TEMP   /(D1*D1)
        PERCE=RC*PRO*ABS(PRO)**.8518
        PERCE=PERCE-T1
        UL=PERCE/COMPR*1000.
        WRITE(IOUT,800) LT,NX1,NX2,C,D1,COMPR,PRO,V,PERCE,UL,T1
150     CONTINUE
 800    FORMAT(1H0,1X,I5,6X,I5,7X,I5,8X,
       1F4.0,4X,F6.0,6X,F6.0,
       12X,F8.1,4X,F7.2,5X,F6.1,2X,F9.1,5X,F6.1)
 1      CONTINUE
        SMAX=0.0
        IF(NORS.EQ.0)GO TO 2021
        DO 2020 I=1,NORS
        K=NOJBC(I)
```

BEST COPY AVAILABLE

```
          IF(K.LE.0.OR.K.GT.700) WRITE(IOUT,9001) I,K
9001      FORMAT(1X,2I5)
      READ(4'K) NOXYT,(COR1(I1),I1=1,5),RC,DEMAND
       DMD=DEMAND
       ABD=ABS(DMD)
       HGL=COR4(I)
       S=ABS(HGL-HDGL(K))
       IF(S.GT.SMAX) SMAX=S
2020   CONTINUE
2021     CONTINUE
       IF(NL.EQ.0) GO TO 3317
       DO 1712 I=1,N
      DELP(I)=0.
1712   CONTINUE
       DO 1710 I=1,LINES
       L1=NDSD1(I)
       L2=NDSD1(I+1)-1
       IF(L1.GT.L2) GO TO 1710
       RC=QTJU(I)
       DO 1711 L=L1,L2
       J=NDLAL(L)
       S=1.0
       IF(J.LT.0) S=-1.0
       J=IABS(J)
      DELP(J)=DELP(J)+S*RC
1711   CONTINUE
1710   CONTINUE
      IF(NORS.EQ.0) GO TO 2453
      DO 2454 I=1,NORS
      J=NIL(I)
     DELP(J)=0.
2454   CONTINUE
2453   CONTINUE
      DO 1713 I=1,N
     S=ABS(DELP(I))
      IF(S.GT.SMAX) SMAX=S
1713   CONTINUE
3317   WRITE(IOUT,2173) SMAX
2173   FORMAT(1H1,    29HMAXIMUM UNBALANCED HEAD LOSS=,E12.5)
      WRITE(IOUT,9000)
9000   FORMAT(1H1)
      RETURN
      END
```

MICROFILMED FROM BEST
AVAILABLE COPY

```
      SUBROUTINE INPUT(NSTOP)
      COMMON COR4S(20),SMIN,SMAX,DINCS,DINCL,COR4(20),DELP(200),
     2COB1(20),COB2(20),COB3(20),COB0(20),A(6400),MAXQ,D,
     3RP,READ,RP1,RP2,RP3,RHO,METI,DIAT,DMAX,
     4NOLAL(1200),J,K,JJ,IOUT,IIN,NRES,NOJUC(20),NRV(20),NRV1(20),
     5NIG(20),NOSD1(402),NLNJ(401),LNJ(800),
     6NOJT(20),NRN(20),N,LINES,JUNCS,NO1,NO2,NOPT,
     7IDEX,NORS,NTM,ITM1,ITM2,LPKT,NL,ITJ1,ITJ2,KKK,NOBP,
     8NR,NOD1,L1,L2,L3,KK,LP,LNTH,NOLJI,IRIT,IMETER,IOUT2,
     9MAXIT,IC,IC1,IFIRST,NOPRV,TC1S
      EQUIVALENCE(NOLJ(1),NOSD(2),NOSD1(3)),(NOPIP(1),LPRP(1),NOLAL(1))
     1,(NOLAL(401),NLJJ(1))
      EQUIVALENCE (A(1),OTJU(1),IP(1)),(A(401),NLT(1),IDL(1)),
     1(A(801),ILI(1),IPL(1)),(A(1201),LLIN(1),NNST(1)),
     2(A(1601),JUNCL(1)),(A(2001),IL(1)),(A(2401),HDGL(1),NX(1))
     3,(A(2801),INDK(1)),(A(4001),NUM(1)),(A(5201),ILOP(1))
     4,(A(5401),NOST(1)),(A(5801),NOST1(1))
      DIMENSION NOLJ(400),NOSD(401),OTJU(400),HDGL(400),NOST1(200),
     1IP(400),IDL(400),NLT(400),ILI(400),IPL(400),LLIN(400),NNST(400),
     2NLJJ(400),JUNCL(400),IL(400),NX(400),LPRP(400),ISAVE(20),
     3NOPIP(400),INDK(1200),NUM(1200),ILOP(200),NOST(400)
      REAL MAXQ,COM(20),METI
      LDIM=400
      IF(IFIRST.EQ.2) GO TO 410
      IF(IC1.LE.0) GO TO 411
      GO TO 410
411   READ(IIN,2,END=3) COM
      NSTOP=0
      WRITE(IOUT,5) COM
2     FORMAT(20A4)
5     FORMAT(1H0,20A4)
      IDEX=1
      READ(IIN,6) MAXQ,PERCE,MAXIT,IRIT,IMETER,SMIN,SMAX
     1,DINCS,DINCL,DIAT,DMAX,NOPT
6     FORMAT(2F6.0,I4,2I2,6F6.0,I2)
      METI=1.
      IF(IMETER.EQ.1) METI=1081.72
      IF(IMETER.EQ.2) METI=1.4037
      NRES=0
      JUNCS=0
      SUM=0.
      J=0
C READ JUNCTION DATA
      DO 12 I=1,1000
      READ(IIN,13,ERR=600) NOXYT,(A(II),II=1,5),RC,D,TEMP
13    FORMAT(I5,5A4,3F7.0)
      IF(NOXYT.LT.1) GO TO 15
      IF(D.EQ.-1.) GO TO 38
      D=D*PERCE
      SUM=SUM+D
      GO TO 300
```

2-2-18

MICROFILMED FROM BEST
AVAILABLE COPY

```
 38.    NRES=NRES+1
        NRN(NRES)=I
300     JUNCS=JUNCS+1
       WRITE(4'I) NOXYT,(A(I1),I1=1,5),RC,D
        NLNJ(I)=NOXYT
        IF(TEMP.LT.1.) GO TO 12
        J=J+1
        NOJT(J)=1
        COR4S(J)=TEMP
 12     CONTINUE
 15       CONTINUE
        DO 9 I=1,LDIM
        NOLJ(I)=0
 9      CONTINUE
        NOPRV=J
         RC=SUM/FLOAT(NRES)
        DO 301 I=1,NRES
        L=NRN(I)
       READ(4'L) NOXYT,(A(I1),I1=1,5),EL,D
       D=-RC
       WRITE(4'L) NOXYT,(A(I1),I1=1,5),EL,D
 301    CONTINUE
        LINES=0
       Q=0.
        T1=0.
C READ LINE DATA
        DO 21 L=1,1000
        READ(TIN,14,ERR=602) LT,NO1,NO2,COMPR,D,C,EQL,NEW
 14     FORMAT(3I5,4F6.0,I5)
        IF(LT.LT.1) GO TO 25
        LINES=LINES+1
        COEF=10.41/(C*D**2.63)**1.8518*MET1*(COMPR+EQL)
        DO 29 M1=1,JUNCS
        IF(NLNJ(M1)-NO1) 29,33,29
 29     CONTINUE
         WRITE(IODT,17) LT,NO1
        IDEX=0
 33     DO 35 M2=1,JUNCS
        IF(NLNJ(M2)-NO2) 35,37,35
 35     CONTINUE
         WRITE(IOUT,17) LT,NO2
        IDEX=0
 17     FORMAT(1H ,'MATCH NOT FOUND',2I5)
 37     CONTINUE
        WRITE(8'L) LT,M1,M2,COMPR,D,D,C,T1,COEF,Q,NEW
        NOLJ(M1)=NOLJ(M1)+1
        NOLJ(M2)=NOLJ(M2)+1
 21     CONTINUE
 25     CONTINUE
        WRITE(IOUT2) (NOLJ(J),J=1,LDIM)
        J1=0
        DO 403 I=1,JUNCS
        DO 401 J=1,LINES
```

BEST COPY AVAILABLE

```
      READ(8'J) LT,NO1,NO2
      IF(NO1.EQ.1) GO TO 403
      IF(NO2.EQ.1) GO TO 403
401     CONTINUE
      J1=J1+1
      ISAVE(J1)=I
403   CONTINUE
      IF(J1.EQ.0) GO TO 410
      WRITE(IOUT,3318)
3318  FORMAT(1H0,'THE FOLLOWING JUNCTIONS ARE NOT CONNECTED')
      DO 420 J=1,J1
      I=ISAVE(J)
      WRITE(IOUT,3319) NLNJ(I)
3319  FORMAT(1H0,I8)
420     CONTINUE
410   CONTINUE
      REWIND IOUT2
      READ(IOUT2) (NOLJ(J),J=1,LDIM)
      IF(IDEX.EQ.0) GO TO 18
      IF(IRIT.GT.0) GO TO 16
18    CONTINUE
      DO 11 I=1,LINES
      READ(8'I) LT,NO1,NO2
11    WRITE(IOUT,10) LT,NLNJ(NO1),NLNJ(NO2)
      IF(IDEX.EQ.0) STOP 'ERROR IN DATA'
16    CONTINUE
10    FORMAT(10 ,3I5)
      NHUM=JUNCS+1
      DO 7 I=1,NHUM
7     NLNJ(I)=0
      NHUM=LINES*2
      DO 8 I=1,NHUM
8     LNJ(I)=0
      NLNJ(1)=1
      NTM=1
      J=1
      JUNCS=JUNCS+1
      DO 200 I=2,JUNCS
      NTM=NTM+NOLJ(J)
      J=I
200   NLNJ(I)=NTM
      JUNCS=JUNCS-1
      Q=0.
      DO 201 I=1,LINES
      READ(8'I) LT,NO1,NO2,COMPR,D1,DO,C,T1,COEF,Q,NEW
      WRITE(8'I) LT,NO1,NO2,COMPR,D1,DO,C,T1,COEF,Q,NEW
      IF(NO1.EQ.0) GO TO 201
      ITM1=NLNJ(NO1)
      ITM2=NLNJ(NO2)
      NLNJ(NO1)=ITM1+1
      NLNJ(NO2)=ITM2+1
```

BEST COPY AVAILABLE

```
      LNJ(ITM1)=I
      LNJ(ITM2)=I
201   CONTINUE
      DO 203 I=1,JUNCS
203   NLNJ(I)=NLNJ(I)-NOLJ(I)
      WRITE(IOUT,305) JUNCS,LINES
305   FORMAT(1H0,'NO OF JUNCTIONS = ',I5,'  NO OF LINES = ',I5)
      WRITE(IOUT,306) SUM
306   FORMAT(1H0,'TOTAL SYSTEM FLOW = ',F10.1,' GPM OR CMH')
      RETURN
3     NSTOP=1
      RETURN
600   WRITE(IOUT,601) NOXYT
601   FORMAT(' CONVERSION ERROR IN DATA NEAR JUNCTION ',I5)
      NSTOP=1
      RETURN
602   WRITE(IOUT,603) LT
603   FORMAT(' CONVERSION ERROR IN DATA NEAR LINE ',I5)
      NSTOP=1
      END
```

BEST COPY AVAILABLE

```
C SUBROUTINE NEEDED
C  INPUT
C  TREE
C  LOOP
C  PARAM
C  NUMBER
C  MATRIX
C  PROFIL
C  HYDGL
C  ZERO
C  CHNG
C  COMP
C  DIA
C  COMPAR
      COMMON CORAS(20),SMIN,SMAX,DINCS,DINCL,COR4(20),DELP(200),
     2COB1(20),COB2(20),COB3(20),COB0(20),A(6400),MAXQ,D,
     3RP,HEAD,RP1,RP2,RP3,RHD,METI,DIAT,DMAX,
     4NOLAL(1200),J,K,JJ,IOUT,IIN,NRES,NOJUC(20),NRV(20),NRV1(20),
     5NIL(20),NOSD1(402),NLNJ(401),LNJ(800),
     6NOJT(20),NRN(20),N,LINES,JUNCS,NO1,NO2,NOPT,
     7IDEX,NORS,NTM,ITM1,ITM2,LPRT,NL,IIJ1,ITJ2,KKK,NOBP,
     8NR,NODJ,L1,L2,L3,KK,LP,LNTH,NOLJ1,IRIT,IMETER,IOUT2,
     9MAXIT,IC,IC1,IFIRST,NOPRV,IC1S
      EQUIVALENCE(NOLJ(1),NOSD(2),NOSD1(3)),(NOP1P(1),LPRP(1),NOLAL(1))
     1,(NOLAL(401),NLJJ(1))
      EQUIVALENCE (A(1),OTJU(1),IP(1)),(A(401),NLT(1),IDL(1)),
     1(A(801),IGI(1),IPL(1)),(A(1201),LLIN(1),NNST(1)),
     2(A(1601),JUMCL(1)),(A(2001),IL(1)),(A(2401),HDGL(1),NX(1))
     3,(A(2801),INDK(1)),(A(4001),NUM(1)),(A(5201),ILOP(1))
     4,(A(5401),NOFT(1)),(A(5801),NOST1(1))
      DIMENSION NOLJ(400),NOSD(401),OTJU(400),HDGL(400),NOST1(200),
     1IP(400),IDL(400),NLT(400),IGI(400),IPL(400),LLIN(400),NNST(400),
     2NLLJ(400),JUMCL(400),IL(400),NX(400),LPRP(400),ISAVE(20),
     3NOP1P(400),INDE(1200),NUM(1200),ILOP(200),NOST(400)
      REAL MAXQ,MPT1
      IC1=0
      IOUT=6
      IIN=5
      IC1S=1
      CALL ASSIGN(4,'TEMP',4,'SCR',NC,1)
      DEFINE FILE 4(1000,15,U,NREC)
      TYPE 414
414   FORMAT(' ENTER SAVE FILE NAME'/)
      CALL ASSIGN(9,SAVE,-23,'NEW',NC,1)
      IOUTZ=3
      TYPE 412
412   FORMAT(' ENTER DATA FILE NAME'/)
      CALL ASSIGN(IIN,FILE,-23)
      TYPE 413
413   FORMAT(' ENTER DATA OUTPUT FILE NAME'/)
      CALL ASSIGN(IOUT,OUTPUT,-23)
```

14-15

BEST COPY AVAILABLE

```
      CALL ASSIGN(10012,'TEMP1',5,'SCR',NC,1)
      CALL ASSIGN(8,'TEMP2',5,'SCR',NC,1)
      DEFINE FILE 8(1000,18,U,NREC)
       IFIRST=1
1     CONTINUE
       GO TO 3
2     CALL CHNG
3      CALL INPUT(NSTOP)
      IF(NSTOP.EQ.1) GO TO 5
      CALL TREE
      CALL LOOP
      CALL PARAM
       IF(LPKT.NE.0) GO TO 315
      CALL HYDGL
      CALL COMPAR
      CALL PROFIL
      STOP 'TREE NETWORK'
315    CONTINUE
      CALL NUMBER
      CALL MATRIX
       IF(IC1.GE.IC1S) GO TO 2
      CALL HYDGL
      IF(NOBP.EQ.0) GO TO 316
      IF(IFIRST.NE.1) GO TO 316
      IF(NOPRV.NE.0) CALL ZERO
      IFIRST=2
      IF(IC1.EQ.1) GO TO 1
      IC1=1
      IF(NOBP.GT.NOPRV) GO TO 1
316   CALL COMPAR
      CALL PROFIL
       GO TO 1
5     STOP 'END OF RUN'
       END
```

BEST COPY AVAILABLE

```
      SUBROUTINE TREE
     COMMON COR45(20),SMIN,SMAX,DINCS,LINCL,COR4(20),DELP(200),
    2COB1(20),COB2(20),COB3(20),COB0(20),A(6400),MAXQ,D,
    3RP,HEAD,RP1,RP2,RP3,RHO,MET1,DIAT,DMAX,
    4NOLAL(1200),J,K,JJ,IOUT,IIN,NRES,NOJUC(20),NRV(20),NRV1(20),
    5NIL(20),NOSD1(402),NLNJ(401),LNJ(800),
    6NOJT(20),NRN(20),N,LINES,JUNCS,NO1,NO2,NOPT,
    71DEX,NORS,NTM,ITM1,ITM2,LPKT,NL,ITJ1,ITJ2,KKK,NOBP,
    8NR,NODT,L1,L2,L3,KK,LP,LNTH,NOLJ1,IRIT,IMETER,IOUT2,
    9MAXIT,IC,IC1,IFIRST,NOPRV,ICIS
      EQUIVALENCE(NOLJ(1),NOSD(2),NOSD1(3)),(NOPIP(1),LPRP(1),NOLAL(1))
    1,(NOLAL(401),NLJJ(1))
      EQUIVALENCE (A(1),QTJU(1),LP(1)),(A(401),NLT(1),IDL(1)),
    1(A(801),ILI(1),IPL(1)),(A(1201),LLIN(1),NNST(1)),
    2(A(1601),JUNCL(1)),(A(2001),IL(1)),(A(2401),HDGL(1),NX(1))
    3,(A(2801),INDK(1)),(A(4001),NOK(1)),(A(5201),ILOP(1))
    4,(A(5401),NOST(1)),(A(5601),NOST1(1))
      DIMENSION NOLJ(400),NOSD(401),QTJU(400),HDGL(400),NOST1(200),
    1LP(400),IDL(400),NLI(400),ILI(400),IPL(400),LLIN(400),NNST(400),
    2NLJJ(400),JUNCL(400),IL(400),NX(400),LPRP(400),ISAVE(20),
    3NOPIP(400),INDK(1200),NOM(1200),ILOP(200),NOST(400)
      REAL MAXQ,C(5),MET1
      DO 16 I=1,LINES
16    ILI(I)=1
      DO 14 I=1,JUNCS
      NLT(I)=0
14    NX(I)=1000
      NX(JUNCS)=0
      K=0
      L2=JUNCS
      IF(NOLJ(JUNCS).EQ.0) GO TO 3315
      NTM=JUNCS-1
      DO 20 L=1,NTM
      IDEX=NLNJ(L2)
      L1=IDEX+NOLJ(L2)-1
      DO 18 I=IDEX,L1
      LP=LNJ(I)
     READ(8'LP) LT,NU,ND,COMPR,D1,D0,DW,T1,COEF,Q,HEW
     NO1=NU
      IF(NO1.EQ.L2) NO1=ND
      J=NX(NO1)
      IF(J.EQ.0) GO TO 18
      IF(J.NE.1000) GO TO 19
      K=K+1
      NX(NO1)=K
      NOPIP(K)=LP
     QTJU(K)=COEF
      GO TO 18
19    RC=COEF
      IF(QTJU(J).LE.RC) GO TO 18
      QTJU(J)=RC
      NOPIP(J)=LP
```

BEST COPY AVAILABLE

```
18      CONTINUE
        TEMP=1.E20
        IF(K.EQ.0) GO TO 3315
        DO 17 I=1,K
        RC=QTJU(I)
        IF(RC.GE.TEMP) GO TO 17
        TEMP=RC
        IM=I
17      CONTINUE
        LP=NOPIP(IM)
       READ(8'LP) LT,NO1,NO2
        L1=NO1
        IF(NX(NO1).NE.0) L1=NO2
         L2=NO1+NO2-L1
        ILI(LP)=0
        NLT(L1)=NLT(L1)+1
        NLT(L2)=1
        NX(L2)=0
        IF(K.EQ.IM) GO TO 21
        LP=NOPIP(K)
       READ(8'LP) LT,NU,ND
       NO1=NU
       IF(NX(NO1).EQ.0) NO1=ND
        NX(NO1)=IM
        NOPIP(IM)=LP
        QTJU(IM)=QTJU(K)
21      K=K-1
20      CONTINUE
         WRITE(IOUT,2024)
2024     FORMAT(' CALCULATE INITIAL FLOWS')
          DO 1918 I=1,JUNCS
       READ(4'I) NOXT,(C(I1),I1=1,5),EL,DEMAND
1918   QTJU(I)=DEMAND
        K=0
        DO 23 I=1,JUNCS
        IF(NLT(I).NE.1) GO TO 23
        K=K+1
        NOST(K)=I
23      CONTINUE
        DO 25 L=1,NTM
29      L1=NOST(K)
        K=K-1
        IDEX=NLNJ(L1)
        J=IDEX+NOLJ(L1)-1
        DO 26 I=IDEX,J
        LP=LNJ(I)
        IF(ILI(LP).EQ.0) GO TO 27
26      CONTINUE
        GO TO 29
27      ILI(LP)=1
       READ(8'LP) LT,NO1,NO2,COMPR,D1,DD,HW,T1,COEF,Q,NEW
         L2=NO1
```

BEST COPY AVAILABLE

```
      IF(L2.EQ.L1) L2=NO2
      J=NLT(L2)-1
      NLT(L2)=J
      IF(J.NE.1) GO TO 28
      K=K+1
      NOST(K)=L2
 28   FLO=QTJU(L1)
      T=FLO
      IF(NO1.EQ.L1) FLO=-FLO
     WRITE(8'LP) LT,NO1,NO2,COMPR,D1,DO,HW,T1,COEF,FLO,NEW
      QTJU(L2)=QTJU(L2)+T
25    CONTINUE
      WRITE(IOUT,367) QTJU(L2)
 367  FORMAT(' NETWORK FLOW BALANCE =',F10.3)
      RETURN
3315  WRITE(IOUT,36)
36    FORMAT(1H0,' NETWORK NOT CONTINUOUS')
     STOP 'STOPPED IN LOOP'
      END
```

BEST COPY AVAILABLE

```
      SUBROUTINE HYDGL
      COMMON COR4S(20),SMIN,SMAX,DINCS,DINCL,COR4(20),DELP(200),
     2COB1(20),COB2(20),COB3(20),COB0(20),A(6400),MAXQ,D,
     3RP,HEAD,RP1,RP2,RP3,RHD,METI,DIAT,DMAX,
     4NOLAL(1200),J,K,JJ,IOUT,IIN,NRES,NOJUC(20),NRV(20),NRV1(20),
     5NIL(20),NOSD1(402),NLNJ(401),LNJ(800),
     6NOJT(20),NRN(20),N,LINES,JUNCS,NO1,NO2,NOPT,
     7IDEX,NORS,NTM,ITM1,ITM2,LPKT,NL,ITJ1,ITJ2,KKK,NOBP,
     8NR,NODI,L1,L2,L3,KK,LP,LNTH,NOLJ1,IRIT,IMETER,IOUT2,
     9MAXIT,IC,IC1,IFIRST,NOPRV,IC1S
      EQUIVALENCE(NOLJ(1),NOSD(2),NOSD1(3)),(NOPIP(1),LPRP(1),NOLAL(1))
     1,(NOLAL(401),NLJJ(1))
      EQUIVALENCE (A(1),QTJU(1),IP(1)),(A(401),NLT(1),IDL(1)),
     1(A(801),ILI(1),IPL(1)),(A(1201),LLIN(1),NNST(1)),
     2(A(1601),JUNCL(1)),(A(2001),IL(1)),(A(2401),HDGL(1),NX(1))
     3,(A(2801),INDK(1)),(A(4001),NUM(1)),(A(5201),ILOP(1))
     4,(A(5401),NOST(1)),(A(5801),NOST1(1))
      DIMENSION NOLJ(400),NOSD(401),QTJU(400),HDGL(400),NOST1(200),
     1IP(400),IDL(400),NLT(400),ILI(400),IPL(400),LLIN(400),NOST(400),
     2NLJJ(400),JUNCL(400),IL(400),NX(400),LPRP(400),ISAVE(20),
     3NOPIP(400),INDK(1200),NUM(1200),ILOP(200),NOST(400)
      REAL MAXQ,METI
      DO 150 I=1,LINES
      READ(8'I) LT,NO1,NO2,COMPR,D1,E0,C,T1,RC,PRQ,NEW
      IF(NO1.EQ.0) GO TO 150
      PERCE=RC*PRQ*ABS(PRQ)**.6518
      PERCE=PERCE-T1
      QTJU(I)=PERCE
150   CONTINUE
      D=D/RP
      ABD=ABS(D)
      HGL=HEAD*(1.0+D*(RP1+ABD*(RP2+ABD*RP3)))+RHD
      HDGL(NR)=HGL
      NOST(1)=NR
      K=1
      L=1
      J=1
      DO 140 I=1,JUNCS
140   IDL(I)=0
      IDL(NR)=1
144   NO1=NOST(K)
      K=K+1
      L1=NLNJ(NO1)
      L2=NLNJ(NO1+1)-1
      DO 141 I=L1,L2
      LP=LNJ(I)
      READ(6'LP) LT,NO,AD
      NO2=NO
      AJJ=-1.0
      IF(NO1.NE.NO2) GO TO 143
      NO2=NO
      AJJ=1.0
```

BEST COPY AVAILABLE

```
 143    IF(IDL(NO2).EQ.0) GO TO 142
1141    CONTINUE
141    CONTINUE
        IF(J.LT.JUNCS) GO TO 144
       GO TO 145
 142    IDL(NO2)=1
        L=L+1
        NOST(L)=NO2
        HDGL(NO2)=HDGL(NO1)-AJJ*QTJU(LP)
       J=J+1
        GO TO 1141
 145    CONTINUE
        RETURN
        END
```

BEST COPY AVAILABLE

```
      SUBROUTINE NUMBER
      COMMON COR45(20),SMIN,SMAX,DINCS,DINCL,COR4(20),DELP(200),
     2COB1(20),COB2(20),COB3(20),COB0(20),A(6400),MAXQ,D,
     3RP,HEAD,RP1,RP2,RP3,RHO,MET1,DIAT,DMAX,
     4NOLAL(1200),J,K,JJ,IOUT,IIN,NRES,NOJUC(20),NRV(20),NRV1(20),
     5NIL(20),NOSD1(402),NLNJ(401),LNJ(800),
     6NOJT(20),NRV(20),N,LINES,JUNCS,NO1,NO2,NOPT,
     7IDEX,NORS,NTM,ITM1,ITM2,LPKT,ML,ITJ1,ITJ2,KKK,NOBP,
     8NR,NODI,L1,L2,L3,KK,LP,LNTH,NOLJ1,IRIT,IMETER,IOUT2,
     9MAXIT,IC,IC1,IFIRST,NOPRV,IC15
      EQUIVALENCE(NOLJ(1),NOSD(2),NOSD1(3)),(NOPIP(1),LPRP(1),NOLAL(1))
     1,(NOLAL(401),NLJJ(1))
      EQUIVALENCE (A(1),QTJU(1),IP(1)),(A(401),NLT(1),IDL(1)),
     1(A(801),LLI(1),IPL(1)),(A(1201),LLIN(1),NNST(1)),
     2(A(1601),JUNCL(1)),(A(2001),IL(1)),(A(2401),HDGL(1),NX(1))
     3,(A(2801),INDK(1)),(A(4001),NOM(1)),(A(5201),ILOP(1))
     4,(A(5401),NOST(1)),(A(5801),NOST1(1))
      DIMENSION NOLJ(400),NOSD(401),QTJU(400),HDGL(400),NOST1(200),
     1IP(400),IDL(400),NLT(400),LLI(400),IPL(400),LLIN(400),NNST(400),
     2NLJJ(400),JUNCL(400),IL(400),NX(400),LPRP(400),ISAVE(20),
     3NOPIP(400),INDK(1200),NOM(1200),ILOP(200),NOST(400)
      REAL MAXQ,C(5),MET1
      WRITE(IOUT,1)
 1    FORMAT(1H ,'NUMBER LOOPS')
      KK=KKK
      JJ=KK
      LMAX=9000
      NODI=0
 94   DO 69 J=1,N
 69   IL(J)=0
      K=1
      IF(JJ.EQ.0) GO TO 77
      I=IPL(JJ)
      IL(I)=1
      IP(1)=1
      L=1
 73   NLP=IP(K)
      L1=NLT(NLP)
      L2=NLT(NLP+1)-1
      IF(L1.GT.L2) GO TO 70
      J=0
      DO 71 I=L1,L2
      LP=INDK(I)
      IF(IL(LP).NE.0) GO TO 71
      J=J+1
      NOST(J)=LP
 71   CONTINUE
      IF(J.NE.0) GO TO 72
 70   K=K+1
      IF(K.LE.L) GO TO 73
      IF(K.EQ.NOLJ1) GO TO 88
 77   IF(KKK.NE.0) GO TO 76
```

BEST COPY AVAILABLE

```
      DO 1432 I=1,4
      IF(IL(I).EQ.0) GO TO 81
1432  CONTINUE
76    I=IPL(KKK)
      KKK=KKK-1
      IF(IL(I).NE.0) GO TO 77
81    L=K
      IP(K)=I
      IL(I)=K
      GO TO 73
72    IF(J.EQ.1) GO TO 82
      DO 83 I=1,J
      LP=NOST(I)
      L1=NLT(LP)
      L2=NLT(LP+1)-1
      MAX=0
      DO 84 L3=L1,L2
      NO1=INLK(L3)
      IF(IL(NO1).EQ.0) MAX=MAX+1
84    CONTINUE
83    NOST1(I)=MAX
      IM=1
      IM1=1
      DO 85 I=2,J
      MAX=NOST1(IM)
      DO 86 L3=I,J
      ITM1=NOST1(L3)
      IF(ITM1.GE.MAX) GO TO 86
      IM=L3
      MAX=ITM1
86    CONTINUE
      ITM1=NOST1(IM)
      ITM2=NOST(IM)
      NOST1(IM)=NOST1(IM1)
      NOST(IM)=NOST(IM1)
      NOST1(IM1)=ITM1
      NOST(IM1)=ITM2
      IM=I
      IM1=I
85    CONTINUE
82    DO 87 I=1,J
      L=L+1
      NO1=NOST(I)
      IP(L)=NO1
87    IL(NO1)=L
      K=K+1
      GO TO 73
88    IF(LMAX.EQ.0) GO TO 92
      IM1=1
      L1=1
      MAX=0
      DO 95 I=2,N
      L2=NLT(I)-1
```

BEST COPY AVAILABLE

```
      IF(L1.GT.L2) GO TO 91
      LP=IL(IM1)
      DO 96 L=L1,L2
      L3=INDK(L)
      K=IABS(LP-IL(L3))
96    IF(K.GT.MAX) MAX=K
91    IM1=I
95    L1=L2+1
      IF(MAX.GE.LMAX) GO TO 93
      LMAX=MAX
      KMAX=JJ
93    JJ=JJ-1
      KKK=KK
      IF(JJ.GT.0) GO TO 94
      JJ=KMAX
      NODI=LMAX
      LMAX=0
      IF(JJ.GT.1) GO TO 94
92    CONTINUE
     NODI=NODI+1
     LNTH=NODI+N
     WRITE(IOUT,1870) LNTH
1870 FORMAT(1H ,'LENGTH OF VECTOR A = ',I5)
      IF(TRIT.NE.0) GO TO 360
     WRITE(IOUT,1869)
      DO 1867 I=1,N
      K=IP(I)
      L1=ILOP(K)
      L2=ILOP(K+1)-1
1867     WRITE(IOUT,1868)I,(NUM(J),J=L1,L2)
1869 FORMAT(5X,'LOOP              PIPES')
1868  FORMAT(3X,I5,8X,20I5,/18X,20I5)
360   CONTINUE
      ITM1=NOSD1(LINES+1)-1
      DO 110 I=1,ITM1
      LP=NOLAL(I)
      J=IABS(LP)
      J=IL(J)
 110  NOLAL(I)=ISIGN(J,LP)
C TEST FOR LOOPS WITH INITIAL FLOW = 0. IN ALL LINES
C  IN THE LOOP. SET FLOW TO 1. FOR ALL SUCH LINES
      AK=1.0
      L1=1
      DO 1820 I=1,N
      L2=ILOP(I+1)-1
      DO 1821 L=L1,L2
      LP=NUM(L)
      LP=IABS(LP)
     READ(8'LP) LT,NO1,NO2,COMPR,DI,DO,HW,TT,COEF,Q,NEW
      IF(Q.NE.0.0) GO TO 1820
1821  CONTINUE
      IF(L.GT.CL) GO TO 1824
1825  DO 1823 L=L1,L2
```

E.2.31

```
      LP=NDM(L)
      J=IABS(LP)
      READ(8'J) LT,NO1,NO2,COMPR,D1,DO,HW,T1,COEF,Q,NEW
      Q=AK*SIGN(1.0,FLOAT(LP))
1823  WRITE(8'J) LT,NO1,NO2,COMPR,D1,DO,HW,T1,COEF,Q,NEW
      GO TO 1820
C  PSEUDO LOOPS
1824  AK=D/2.0
      D=AK
      K=I-NL
      K=NOJOC(K)
      READ(4'K) NOXYT,(C(II),II=1,5),EL,DEMAND
      DEMAND=DEMAND+D
      WRITE(4'K) NOXYT,(C(II),II=1,5),EL,DEMAND
      GO TO 1825
1820  L1=L2+1
      LK=1
      NL=NL+1
      IF(NL.GT.N) GO TO 1700
      K=1
      DO 1701 I=NL,N
      NTL(K)=IL(I)
      K=K+1
1701  CONTINUE
1700  NL=NL-1
      DO 112 I=1,LINES
      J=NOSD1(I)
      L2=NOSD1(I+1)-1
      L1=J+1
      K=J
115   IF(L1.GT.L2) GO TO 112
      MIN=IABS(NOLAL(J))
      DO 113 L=L1,L2
      IF(MIN.LE.IABS(NOLAL(L))) GO TO 113
      MIN=IABS(NOLAL(L))
      J=L
113   CONTINUE
      KK=NOLAL(K)
      NOLAL(K)=NOLAL(J)
      NOLAL(J)=KK
      K=L1
      J=K
      L1=L1+1
      GO TO 115
112   CONTINUE
      RETURN
      END
```

MICROFILMED FROM BEST
AVAILABLE COPY

```
      SUBROUTINE PARAM
      COMMON COR4S(20),SMIP,SMAX,DINCS,DINCL,COR4(20),DELP(200),
     2COB1(20),COB2(20),COB3(20),COB0(20),A(6400),MAXQ,D,
     3RP,HEAD,RP1,RP2,RP3,RHO,MET1,DIAT,DMAX,
     4NOLAL(1200),J,K,JJ,IOUT,IIN,NRES,NOJUC(20),NRV(20),NRV1(20),
     5NIL(20),NOSD1(402),NLNJ(401),LNJ(800),
     6NOJT(20),NRN(20),N,LINES,JUNCS,NO1,NO2,NOPT,
     7IDEX,NOBS,NIM,ITM1,ITM2,LPKT,NL,ITJ1,ITJ2,KKK,NOBP,
     8NR,NODT,L1,L2,L3,KK,LP,LPTH,NOLJ1,IRIT,IMETER,IOUT2,
     9MAXIT,IC,IC1,IFIRST,NOPRV,IC15
      EQUIVALENCE(NOLJ(1),NOSD(2),NOSD1(3)),(NOPIP(1),LPRP(1),NOLAL(1))
     1,(NOLAL(401),NLJJ(1))
      EQUIVALENCE (A(1),QTJU(1),IP(1)),(A(401),NLT(1),IDL(1)),
     1(A(801),ILI(1),IPL(1)),(A(1201),LLIN(1),NNST(1)),
     2(A(1601),JUNCL(1)),(A(2001),IL(1)),(A(2401),HDGL(1),NX(1))
     3,(A(2801),INDK(1)),(A(4001),NUM(1)),(A(5201),ILOP(1))
     4,(A(5401),NOST(1)),(A(5801),NOST1(1))
      DIMENSION NOLJ(400),NOSD(401),QTJU(400),HDGL(400),NOST1(200),
     1IP(400),IDL(400),NLT(400),ILI(400),IPL(400),LLIN(400),NNST(400),
     2NLJJ(400),JUNCL(400),IL(400),NX(400),LPRP(400),ISAVE(20),
     3NOPIP(400),INDK(1200),NUM(1200),ILOP(200),NOST(400)
      REAL MAXQ,C(5),MET1
      WRITE(IOUT,1)
1     FORMAT(' ','READ RESERVOIR DATA')
      IF(IC1.GT.0) GO TO 20
      DO 30 I=1,JUNCS
      READ(4'T) IP(I),(COB1(II),II=1,5),RC,DEMAND
30    CONTINUE
      READ(IIN,102) NR,L,RHO,HEAD,RP1,RP2,RP3
102     FORMAT(2I5,5F10.0)
      IF(L.EQ.0) L=1
      RP=L
      DO 2 I=1,JUNCS
      IF(IP(I).EQ.NR) GO TO 3
2     CONTINUE
3     NR=I
20    CONTINUE
      DO 1300 I=1,JUNCS
      LLIN(I)=9000
1300  JUNCL(I)=0
      LLIN(NR)=0
      NOST(1)=NR
      K=1
      L=1
1305  NO1=NOST(K)
      K=K+1
      L1=NLNJ(NO1)
      L2=NLNJ(NO1+1)-1
      JJ=LLIN(NO1)
      DO 1301 I=L1,L2
      LP=LNJ(I)
      KK=1
```

MICROFILMED FROM BEST
AVAILABLE COPY

BEST COPY AVAILABLE

```
      READ(8'LP) LT,NU,NO2
      IF(NO2.NE.NO1) GO TO 1306
      NO2=NU
       KK=-1
1306   J=NOLJ(LP)
       JJJ=J+JJ
       IF(JJJ.GE.LLIN(NO2)) GO TO 1301
       LLIN(NO2)=JJJ
       IF(JUNCL(NO2).NE.0) GO TO 1304
       L=L+1
       NOST(L)=NO2
1304   NLJJ(NO2)=KK*LP
       JUNCL(NO2)=NO1
1301   CONTINUE
       IF(K.LE.L) GO TO 1305
       IF(L.EQ.JUNCS) GO TO 1213
       WRITE(IOUT,1214)
       STOP'IN PARAM'
1214   FORMAT(1H1,'SECOND NETWORK NOT CONTINUOUS')
1213   NORS=NRES-1
       ITM1=KKK+NORS
       KKK=KKK+1
       IF(KKK.GT.ITM1) GO TO 2001
       DO 2000 I=KKK,ITM1
2000   IPL(I)=NL+I+1-KKK
2001   KKK=ITM1
       IF(NORS.EQ.0) GO TO 100
       DO 101 I=1,NORS
       IF(IC1.GT.0) GO TO 21
       READ(IIN,102) K,L,COR4(I)
       DO 6 I1=1,JUNCS
       IF(IP(I1).EQ.K) GO TO 7
 6     CONTINUE
      WRITE(IOUT,1700) K
      STOP
1700   FORMAT(' RESERVOIR',I5,' IS NOT NUMBERED CORRECTLY')
 7     K=I1
       NOJUC(I)=K
21     CONTINUE
       K=NOJUC(I)
1310   LP=NLJJ(K)
       K=JUNCL(K)
       NUM(LPKT)=LP
       LPKT=LPKT+1
       IF(K.NE.NR) GO TO 1310
       NOLJ1=NOLJ1+1
       ILOP(NOLJ1)=LPKT
101    CONTINUE
100    N=NL+NORS
       IF(IC1.GT.0) GO TO 1714
       NOBP=0
       DO 31 I=1,LINES
```

BEST COPY AVAILABLE

```
      READ(6'I) IP(I)
31     CONTINUE
       DO 1715 I=1,40
       READ(IIN,102) LPB,L,COB0(I),COB1(I),COB2(I),COB3(I)
        IF(LPB.LT.1) GO TO 1714
        DO 4 II=1,LINES
       IF(IP(II).EQ.LPB) GO TO 5
4     CONTINUE
      WRITE(IOUT,1701) LPB
      STOP
1701  FORMAT(' BOOSTER PUMP',I5,' IS NOT NUMBERED CORECTLY')
5      NRN(I)=II
       NOBP=NOBP+1
1715   CONTINUE
1714   CONTINUE
      READ(4'NR) NOXYT,(C(II),II=1,5),EL,D
       LPKT=LPKT-1
       DO 61 I=1,LINES
61     NOLJ(I)=0
       IF(LPKT.NE.0) GO TO 3314
       RETURN
3314   DO 62 I=1,LPKT
       J=IABS(NUM(I))
62     NOLJ(J)=NOLJ(J)+1
       J=1
       NOSD(1)=1
       DO 63 I=2,LINES
       NOSD(I)=NOSD(J)+NOLJ(J)
63     J=1
       KM1=1
       K=2
       LP=ILOP(2)
       DO 64 I=1,LPKT
       JJ=NUM(I)
       J=IABS(JJ)
       L=NOSD(J)
       NOSD(J)=L+1
       IF(I.LT.LP) GO TO 65
       KM1=K
       K=K+1
       LP=ILOP(K)
65     NOLAL(L)=ISIGN(KM1,JJ)
64     CONTINUE
       NOSD1(1)=1
       LPKT=1
       J=1
       DO 66 I=1,N
       NL1(I)=LPKT
       KPS=LPKT
       K=ILOP(I+1)-1
       DO 66 L=J,K
       LP=IABS(NUM(L))
```

BEST COPY AVAILABLE

```
      L1=NOSD1(LP)
      L2=NOSD1(LP+1)-1
      DO 67 L3=L1,L2
      NO1=IABS(NOLAL(L3))
      IF(NO1.EQ.1) GO TO 67
      IF(KPS.EQ.LPKT) GO TO 9991
      ITM2=LPKT-1
      DO 999 NO2=KPS,ITM2
      IF(INDK(NO2).EQ.NO1) GO TO 67
999   CONTINUE
9991  INDK(LPKT)=NO1
      LPKT=LPKT+1
67    CONTINUE
68    CONTINUE
      J=K+1
66    CONTINUE
      NLT(N+1)=LPKT
      RETURN
      END
```

BEST COPY AVAILABLE

```
      SUBROUTINE LOOP
      COMMON COR4S(20),SMIN,SMAX,DINCS,DINCL,COR4(20),DELP
     2COB1(20),COB2(20),COB3(20),COB0(20),A(6400),MAXQ,Q
     3RP,HEAD,RP1,RP2,RP3,RAD,METI,DIAT,DMAX,
     4NOLAL(1200),J,K,JJ,IOUT,ITN,NRES,NOJUC(20),NRV(20),NRV
     5NIL(20),NOSD1(402),NLNJ(401),LNJ(800),
     6NOJT(20),NRN(20),N,LINES,JUNCS,NO1,NO2,NOPT,
     7IDEX,NORS,NTM,ITM1,ITM2,LPKT,NL,ITJ1,ITJ2,KKK,NOBP,
     8NR,NODT,L1,L2,L3,KK,LP,LNTH,NOLJ1,IRIT,IMETER,IOUT2,
     9MAXIT,IC,IC1,IFIRST,NOPRV,IC1S
      EQUIVALENCE(NOLJ(1),NOSD(2),NOSD1(3)),(NOPIP(1),LPRP(1),NOLAL(1))
     1,(NOLAL(401),NLJJ(1))
      EQUIVALENCE (A(1),QTJU(1),IP(1)),(A(401),NLT(1),IDL(1)),
     1(A(801),ILI(1),IPL(1)),(A(1201),LLIN(1),NNST(1)),
     2(A(1601),JUNCL(1)),(A(2001),IL(1)),(A(2401),HDGL(1),NX(1)),
     3,(A(2801),INDK(1)),(A(4001),NUM(1)),(A(5201),ILOP(1))
     4,(A(5401),NOST(1)),(A(5801),NOST1(1))
      DIMENSION NOLJ(400),NOSD(401),QTJU(400),HDGL(400),NOST1(200),
     1IP(400),IDL(400),NLT(400),ILI(400),IPL(400),LLIN(400),NNST(400),
     2NLJJ(400),JUNCL(400),IL(400),NX(400),LPRP(400),ISAVE(20),
     3NOPIP(400),INDK(1200),NUM(1200),ILOP(200),NOST(400)
      REAL MAXQ,METI
      WRITE(IOUT,1)
    1 FORMAT(1H ,'FIND LOOPS')
      DO 2 I=1,LINES
      LPRP(I)=1
    2 CONTINUE
      K=0
      I=3
      DO 55 JTR=1,JUNCS
      IF(NOLJ(JTR).NE.1) GO TO 54
      ITJ1=JTR
      NOLJ(ITJ1)=0
      GO TO 60
   52 CONTINUE
   54 IF(NOLJ(JTR).NE.2) GO TO 55
      K=K+1
      NOST(K)=JTR
   55 CONTINUE
      NL=LINES-JUNCS+1
      LPKT=1
      KKK=0
      IF(NL) 33,33,3316
 3316 DO 51 NLP=1,NL
      ILOP(NLP)=LPKT
 4114 DO 59 I=1,JUNCS
   59 JUNCL(I)=0
   53 IF(K.NE.0) GO TO 34
      NOL1=1000
      DO 35 I=1,JUNCS
      IF(NOLJ(I).EQ.0.OR.NOLJ(I).GE.NOL1) GO TO 35
      L1=1
```

BEST COPY AVAILABLE

```
      NDL1=NOLJ(1)
35    CONTINUE
      ITJ2=L1
      NOLJ(L1)=NDL1-1
      GO TO 1435
34    L1=NOST(K)
      K=K-1
      IF(NOLJ(L1).NE.2) GO TO 53
      NOLJ(L1)=0
1435  IDEX=NLNJ(L1)
57    LP=LNJ(IDEX)
      ITT=1
      IDEX=IDEX+1
      IF(LPRP(LP).NE.0) GO TO 56
      GO TO 57
56    LPRP(LP)=0
      KM1=0
     READ(8'LP) LT,NO1,NO2
     ITJ1=NO2
      J=1
      IF(ITJ1.NE.L1) GO TO 58
      J=-1
     ITJ1=NO1
58    NUM(LPKT)=J*LP
      LPKT=LPKT+1
      NOLJ(ITJ1)=NOLJ(ITJ1)-1
      IF(NOLJ(L1).NE.0) GO TO 1436
37    LP=LNJ(IDEX)
      ITT=2
      IDEX=IDEX+1
      IF(LPRP(LP).EQ.0) GO TO 37
      LPRP(LP)=0
     READ(8'LP) LT,NO1,NO2
     ITJ2=NO1
      J=1
      IF(ITJ2.NE.L1) GO TO 41
     ITJ2=NO2
      J=-1
41    NUM(LPKT)=J*LP
      LPKT=LPKT+1
      NOLJ(ITJ2)=NOLJ(ITJ2)-1
      IF(ITJ1.EQ.ITJ2) GO TO 4115
1436  NNST(1)=ITJ1
      KK=1
      KNT=1
      JUNCL(ITJ1)=L1
39    NO1=NNST(KK)
      KK=KK+1
      ITM1=NLNJ(NO1)
      ITM2=NLNJ(NO1+1)-1
      DO 40 I=ITM1,ITM2
      LP=LNJ(I)
```

BEST COPY AVAILABLE

```
      ITI=3
      IF(LPRP(LP).EQ.0) GO TO 40
      READ(6'LP) LT,NO,ND
      NO2=NO
      J=-1
      IF(NO2.NE.NO1) GO TO 42
      J=1
      NO2=ND
42    IF(JUNCL(NO2).NE.0) GO TO 40
      JUNCL(NO2)=NO1
      KNT=KNT+1
      NNST(KNT)=NO2
      NLJJ(NO2)=LP*J
40    CONTINUE
      IF(JUNCL(ITJ2).NE.0) GO TO 36
      IF(KK.LE.KUT) GO TO 39
      LPKT=LPKT-2
      IF(NOBJ(L1).NE.0) LPKT=LPKT+1
      I=4
      GO TO 49
43    ITJ1=ITJ2
      I=5
      GO TO 49
44    GO TO 4114
38    J=ITJ2
4113  LP=NLJJ(J)
      IF(NLNJ(J+1)-NLNJ(J).EQ.2) KM1=1
      J=JUNCL(J)
      NLP(LPKT)=LP
      LPKT=LPKT+1
      IF(J.NE.ITJ1) GO TO 4113
      IF(NLNJ(L1+1)-NLNJ(L1).EQ.2.OR.LPKT-LLOP(NLP).GT.7) KM1=1
      IF(KM1.EQ.0) GO TO 4115
      KKK=KKK+1
      IPL(KKK)=NLP
4115  I=1
      GO TO 49
62    CONTINUE
51    CONTINUE
      GO TO 33
49    J=NOBJ(ITJ1)
      IF(J.NE.2) GO TO 45
      K=K+1
      NOST(K)=ITJ1
      GO TO 46
45    IF(J.NE.1) GO TO 46
      NOBJ(ITJ1)=0
60    IDEX=NLNJ(ITJ1)
48    LP=LNJ(IDEX)
      ITI=4
      IF(LPRP(LP).NE.0) GO TO 47
      IDEX=IDEX+1
```

```
      GO TO 48
47    LPRP(LP)=0
     READ(8'LP) LT,NO,ND
     L1=NO
      IF(L1.EQ.ITJ1) L1=ND
      ITJ1=L1
      NOLJ(ITJ1)=NOLJ(ITJ1)-1
      GO TO 49
48    GO TO (50,62,52,43,44),I
50    I=Z
      ITJ1=ITJZ
      GO TO 49
33    CONTINUE
      NOLJ1=NL+1
      ILOP(NOLJ1)=LPNT
     RETURN
      STOP'STOPPED IN LOOP'
      END
```

MICROFILMED FROM BEST
AVAILABLE COPY