University of Kentucky
UKnowledge

KWRRI Research Reports

Kentucky Water Resources Research Institute

Minimum Cost Design of Water Distribution Systems

Don J. Wood
University of Kentucky

C. O. Charles
University of Kentucky

Right click to open a feedback form in a new tab to let us know how this document benefits you.

MINIMUM COST DESIGN OF WATER DISTRIBUTION SYSTEMS

Dr. Don J. Wood
Principal Investigator

Graduate Student Assistant: C. O. Charles, Ph. D.

Project Number B-017-KY (Completion Report)
Agreement Number 14-01-0001-3285
Period of Project - July 1970 - June 1973

University of Kentucky Water Resources Institute
Lexington, Kentucky

September, 1973

TABLE OF CONTENTS

ABSTRACT

The objective of this study was to develop the analytical tools and procedures for minimum cost design of water distribution systems. Both analog and digital means of carrying out pressure and flow calculations were developed. As a result of this effort, digital programs for pressure and flow calculations in water distribution systems were written and have been widely distributed to practicing engineers. One procedure is based on a direct solution of the basic system equations using a linearization scheme and has several advantages over conventional techniques such as the Hardy Cross method. These include avoiding the need to initially balance the network and an assured convergence of the procedure.

Using this tool a procedure was developed for selecting pipe diameter which will result in a minimum cost design within the prescribed constraints. The method of steepest ascent and dynamic programming concepts were used to carry out the optimization. This procedure applies to closed loop systems without internal pumping. However, this work provides a basis for extending the concepts to more generalized water distribution systems.

ACKNOWLEDGEMENTS

INTRODUCTION

The objective of the study as outlined in the original proposal was, "to investigate the various functional constraints controlling the design of water distribution systems and develop analytical methods and digital computer routines which can be utilized to design a water distribution system at minimum cost". It was determined early in this investigation that the available means for the hydraulic analysis of water distribution systems did not lend themselves well to a minimum cost analysis. Therefore, a considerable effort was made to develop the analytical tools for pressure and flow calculations which could be incorporated into a minimum cost analysis of water distribution systems.

A promising technique which was investigated was the use of analog simulation for system hydraulics which could be incorporated into a digital-analog scheme to carry out the cost minimization. Techniques for carrying out an analog simulation of pipe system hydraulics on a standard analog computer were developed and reported by the principal investigator (1). It was intended to use the analog computer to model the hydraulics and to use analog to digital conversion and a digital computer to compute the cost. It was felt that with such a model an effective directed search could be undertaken to determine the optimum design of the pipe system. However, it turned out that the necessary equipment for the development of this concept was not available to this project. Since some rather expensive equipment is involved this approach had to be abandoned.

A major effort was devoted to the development of an analytical procedure for hydraulic analysis based on linearization of the basic non-linear system equations. The purpose of developing this approach was that it appeared to offer a method

for handling pipe system hydraulics which could be more easily incorporated into a minimum cost study. A scheme which directly solved the basic equation after linearizing the non-linear terms was developed. A publication is available describing this phase of the study (2). This method of hydraulic analysis refereed to as the linear method offered distinct advantages over the conventional Hardy Cross and Newton Raphson methods which are generally used. Therefore, some additional effort was made developing the linear method for generalized situations. As a result of this, a general computer program has been developed and made available to engineers working in this field. Over one hundred and fifty engineering firms have acquired this program.

Finally, using the linear method for hydraulic analysis a computer program has been developed for minimum cost design of closed loop water distribution systems. This work was primarily the effort of C. O. Charles and is documented in his Ph. D. dissertation (3). In this program the method of steepest ascent and concepts of dynamic programming were employed to formulate a procedure which would select the optimum set of pipe diameters for a closed loop system.

RESEARCH PROCEDURES

The entire project is concerned with the development of the basic elements of an analytical model to be employed for minimum cost design. This entails the conception, formulation and testing of certain analytical procedures. In most cases either analog or digital computer programs were the end product of this effort. The usual procedures for formulating, debugging and testing computer programs were employed.

RESULTS

Analog simulation of pipe system hydraulics

This phase of the investigation has been completely documented in Reference 1 and is available through that publication. As previously stated, however, the necessary digital-analog equipment was not available to develop a technique for optimum design using analog simulation.

Digital programs for the analysis of pipe system hydraulics

A considerable effort was made to develop a digital computer program which would easily handle general water distribution systems in a manner which would lend itself to a minimum cost investigation. This effort resulted in the development of two computer programs. General information pertaining to the development of these programs follows:

The programs will compute steady flow in pipe systems of any arrangement. The system can include pumps, valves, bends, and other minor loss components, storage tanks and source and storage reservoirs. A system of p pipes can be described by the number of junctions, j, the number of closed primary loops, l, and the number of terminal energy points, t, in the system. A junction is simply a point in the system where two or more pipes meet. Any point where flow enters or exits the pipe system is also a junction. A primary loop is a closed loop of pipes in the system which have no other loops within it. A terminal energy point is a point in the system where the fluid energy is known. This is essentially any point where the pressure and the elevation are known. Source or storage reservoirs, pressurized sources, storage tanks and discharge points of known pressure are the most common terminal

3

energy points.   To describe a system the junctions, loops and terminal energy points must be identified.   <u>If a terminal energy point and a junction coincide, this point should be identified as a terminal energy point only.</u>   If the junctions, loops and terminal energy points are identified with the restriction just stated, the following holds for all pipe systems:

$$p = j + \ell + t - 1 \qquad\qquad (1)$$

where

        p = number of pipes

        j = number of junctions

        ℓ = number of loops

        t = number of terminal energy points.

In terms of the unknown discharge in each pipe, a number of continuity and energy equations can be written equaling the number of pipes in the system.   For each junction a continuity equation equating the flow into the junction to the flow out is written as:

$$Q_{in} = Q_{out} \qquad \text{(j equations)} \qquad (2)$$

For each loop the energy equation can be written as follows

$$\Sigma h_L = \Sigma E_P \qquad (\ell \text{ equations}) \qquad (3)$$

where

        h_L = head loss in each pipe (including minor loss)

        E_P = energy put into the liquid by a pump.

If there are no pumps in the loop then the energy equation states that the sum of the head loss around the loop equals zero.

If there are t terminal energy points, t - 1 energy equations can be written for paths between any two terminal energy points as follows

$$\Delta E = \Sigma h_L - \Sigma E_P \qquad (t - 1 \text{ equations}) \qquad (4)$$

where ΔE is the energy difference between the two terminal energy points. Any path in the pipe system can be chosen between the points. However, care must be taken to avoid redundant paths. The best method to avoid this difficulty is to either choose all paths starting at one source (like 1-2, 1-3, 1-4, etc.) or to use the previous end point for a path as the starting point for the next path (like 1-2, 2-3, 3-4, etc.). Either of these methods will result in $t - 1$ equations with no redundant ones.

These junction loop and path equations constitute a set of simultaneous equations equal to the number of pipes in the system which can be solved for the discharge in each pipe. A direct solution of these simultaneous equations is not possible because of the non-linear terms. Two basic methods of solution were considered.

Linear method - For this approach the non-linear terms are linearized giving a set of linear simultaneous equations which can be solved using matrix methods. The linearization is formulated as follows. The line loss is given by:

$$h_{LP} = K_P Q^n \qquad (5)$$

where K_P is a pipe line constant and for the Hazen Williams equation employed in the computer analysis is

$$K_P = \frac{4.73 \, L}{C^{1.852} D^{4.87}} \qquad (6)$$

Here L = line length in ft, D = line diameter in ft and C is the Hazen Williams roughness coefficient. The discharge Q in eqn. 5 is in cfs and the exponent $n = 1.852$.

Minor losses are given by a loss coefficient, M, which multiplies the velocity head to give the loss at the component.

5

This is

$$h_{LM} = M \frac{V^2}{2g} \tag{7}$$

where V is the mean line velocity and g is the gravitational constant. In terms of the discharge this is

$$h_{LM} = K_M Q^2 \tag{8}$$

where

$$K_M = \frac{.02517 \, M}{D^4} \tag{9}$$

The pump head is expressed in two ways.

$$E_P = \frac{Z_P}{Q} \tag{10}$$

For this expression the horsepower put into the system by the pump is given as HP and

$$Z_P = \frac{550 \, HP}{\gamma} \tag{11}$$

where γ = specific weight of the liquid (#/ft^3). Alternately the pump head can be expressed as

$$E_P = A + BQ + CQ^2 \tag{12}$$

where A, B, and C are coefficients of a parabolic characteristic curve which defines the pump operation in the vicinity of the operating point. Since this expression is only valid over a specified range it should not be indiscretely employed in an analysis.

The basic energy equation for a loop or a path between terminal energy points is:

6

$$\Sigma(h_{LP} + h_{LM}) = \Delta E + \Sigma E_P \qquad (13)$$

Here ΔE is the energy difference between the terminal energy points. This equation can be linearized in terms of a flowrate Q_i in the vicinity of the solution. This is done as follows

$$h_{LP} = h_{LPi} + \Delta h_{LP} = K_P Q_i^n + n K_P Q_i^{n-1}(Q-Q_i) \qquad (14)$$

$$h_{LM} = h_{LMi} + \Delta h_{LM} = K_M Q_i^2 + 2 K_M Q_i (Q_i - Q) \qquad (15)$$

$$E_P = E_{Pi} + \Delta E_P = \frac{Z_P}{Q_i} - \frac{Z_P}{Q_i^2}(Q - Q_i) \qquad (16)$$

or:

$$E_P = A + B Q_i + C Q_i^2 + (B + 2C Q_i)(Q - Q_i) \qquad (17)$$

With these substitutions eqn. 13 can be expressed as a linear function of Q as

$$\Sigma(n K_P Q_i^{n-1} + 2 K_M Q_i + \frac{Z_P}{Q_i^2}) Q =$$

$$\Sigma\left(\frac{2 Z_P}{Q_i} + (n-1) K_P Q_i^n + K_M Q_i^2\right) + \Delta E \qquad (18)$$

For the alternate form of the pump head this equation is

$$\Sigma(n K_P Q_i^{n-1} + 2 K_M Q_i - B - 2 C Q_i) Q =$$

$$\Sigma(A - C Q_i^2 + (n-1) K_P Q_i^n + K_M Q_i^2) + \Delta E \qquad (19)$$

Equation 18 (or 19) is employed to formulate an equation for each loop ($\Delta E = 0$) and t - 1 terminal energy equations which combine with

7

the j continuity equations to for a set of P simultaneous linear equations in terms of the flowrate in each pipe.

Path method -   The same notation previously defined in the description of the linear method is used.   The basis of this method is to compute a flow correction ΔQ which when added to an initial set of flowrates (which satisfy continuity) will tend to satisfy the energy equation for each path.   This is

$$\Sigma(h_{LP} + h_{LM}) = \Delta E + \Sigma E_P$$

In terms of the initial flowrate Q_i and the flow correction ΔQ these terms are

$$h_{LP} = h_{LPi} + \Delta h_{LP} = K_P Q_i^n + n K_P Q_i^{n-1} \Delta Q \tag{20}$$

$$h_{LM} = h_{LMi} + \Delta h_{LM} = K_M Q_i^2 + 2 K_M Q_i \Delta Q \tag{21}$$

$$E_P = E_{Pi} + \Delta E_P = \frac{Z_P}{Q_i} - \frac{Z_P}{Q_i^2} \Delta Q \tag{22}$$

or

$$E_P = A + B Q_i + C Q_i^2 + (B - 2 C Q_i) \Delta Q \tag{23}$$

These can be solved to give a flow correction as

$$\Delta Q = \frac{\Delta E - \Sigma(K_P Q_i^n + K_M Q_i^2 - \dfrac{Z_P}{Q_i}}{\Sigma(n K_P Q_i^{n-1} + 2 K_M Q_i + \dfrac{Z_P}{Q_i^2})} \tag{24}$$

or

$$\Delta Q = \frac{\Delta E - \Sigma(K_P Q_i^n + K_M Q_i^2 - (A + B Q_i + C Q_i^2))}{\Sigma(n K_P Q_i^{n-1} + 2 K_M Q_i - B - 2 C Q_i)} \tag{25}$$

8

Using either eqn. 24 or eqn. 25 a flow correction is computed for each path and the flowrates of the pipes in that path are corrected by this amount.

For each method the following information must be available before the hydraulic analysis can be made. For each line the length, diameter and the Hazen Williams Roughness coefficient must be known. This latter parameter is available in handbooks and depends on the type and condition of the pipe. Valves, bends, meters, etc. are included in the analysis by determining the minor loss coefficient for the components. The minor loss coefficient is defined as a constant which multiplies the velocity head in the line to give the head loss at that component. In many cases a standard value for this coefficient is given in various references. This coefficient can also be easily determined if discharge-head loss data is available for the component. Several components can be included in a line by summing their minor loss coefficients.

Pumps can be included in two ways. The useful horsepower (or kilowatts) which the pump puts into the system may be specified. Alternately the coefficients of a parabolic characteristic curve may be specified. This curve represents the pump head-discharge relationship as shown below.
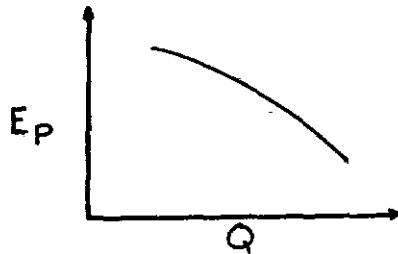
In the normal range of pump operation this relationship can be described closely by

$$E_P = A + BQ + CQ^2$$

9

where A, B, and C are coefficients of the fitted curve. If this representation of the pump is used, however, the solution must yield a discharge in the normal range of operation or the solution will be invalid. This is because the characteristic curve is not valid outside that range.

For each junction the external inflow or outflow is specified and the elevation of the junction is known.

Program description and users options - The programs were developed for use by practicing engineers and were offered to engineers on several bases. Material on the programs has been provided to over 150 engineering firms and individuals. The following brief release provided information to potential users:

Two programs have been developed at the University of Kentucky which will analyze pressure and flow in any pipe system and are available to potential users. These programs are written in FORTRAN IV, G Level and a users guide with source program listings and examples has been prepared. A brief description of the programs follow.

I Program based on linearized system equations -

This program utilized a new procedure for pipe systems analysis which has several advantages over conventional methods. Because this method simultaneously computes the flow in each pipe, the convergence is very fast (usually 3-4 trials to very high accuracy regardless of size of the system). Also, convergence is assured. Initial flowrates are not assumed and changes in flow system demand only require a change in data pertaining to that demand. However, since matrix methods are employed a computer of sufficient storage must be available to use this method for a system of n pipes approximately (n x (n+35) dimensioned storage locations must be available. The IBM 360-65 computer at the University of Kentucky, for example, will handle systems up to 220 pipes with its present storage capacity and without using additional disc storage. The procedure is fast. A 37 pipe system

10

can be analyzed in about 10 seconds while a 125 pipe system takes 2 minutes and 15 seconds on the University of Kentucky computer.

II Program based on loop and path flow adjustments -

This program is essentially an extension of a loop balancing method similar to the Hardy Cross technique to any type of pipe flow system with pumps, valves, etc. included. It does require as input data initial flowrates which satisfy continuity. In rare cases the Hardy Cross procedure does not produce convergence and this situation could occur. However, for most situations the program produces a fast accurate solution. In addition much less storage is required so a large system can be analyzed with a computer of limited storage capacity.

Basic features of both procedures are:

1. Any piping configurations can be analyzed (closed loop networks, tree systems of combinations).

2. Flow units of CFS, GPM, MGD or SI units (M^3/s) can be used.

3. Pump, valves and other lossy components, and storage tanks can be included in the pipe system.

4. Pressures and hydraulic grades at indicated points in the system are output in addition to head changes at pumps, valves and in lines.

5. Data preparation for both programs is straightforward and very similar and allows any number of changes in system parameters (pipe sizes, pump characteristics, flow demands, etc.) to be investigated in a single computer run.

The programs are available to interested users in one of the following ways:

1. Attend two day short course at the University of Kentucky - announcement attached. This is the best means of gaining the necessary experience for using the programs effectively and is especially recommended for persons not presently using computers for hydraulic analysis. All material and computer source programs are provided for participant. Some post-course consultation and

11

a post-course laboratory problem chosen by the participant provide additional aid in implementing the programs.

2. Participate in users course on a correspondence basis. This is primarily for users who have an interest in developing the capability of using the programs but cannot attend a short course. It is desirable that the pariticipant have some background in the use of computers. All material and source programs are provided for the participant in addition to problems. The data for these problems are coded and returned for computer processing. In addition the participant may code and submit data for an additional problem over a period of a year which is of interest to him. Systems of up to 50 pipes will be processed as part of the course (larger systems require a nominal additional charge for computing expenses). Consultation regarding the application of the programs to pipe systems will be provided by phone or mail.

3. Obtain material only. This is primarily for users who are already using a digital computer for hydraulics problems. Users guides, program listings and examples will be sent for both programs.

Complete details of the programs, program listings and examples are provided in the users manuals (4, 5).

Program for optimum design of water distribution networks.

A major effort to develop a programmable procedure for minimum cost design was made. Details of this effort are included in a Ph. D. thesis (3). The salient points of this effort will be covered in this report.

Problem definition - In designing a hydraulic network distribution system, the engineer has not only to meet the demands at particular points in the system, but also should do so within specified constraints and at the least possible cost. For this study the geometrical configuration of the network is prescribed. The cost is a function of diameter and flow and the constraints can be

regarded as of three types: (i) hydraulic (Kirchhoff laws), (ii) pressure and (iii) diameter. There are also different classes of constraints within each type. For example some pressure constraints are of the type that the pressure must be greater than or equal to a minimum while another constraint is that the pressure must not exceed some maximum value. The diameter constraints are normally of two kinds; the first is that no diameter should be less than a certain minimum, the other that the diameters should be available on the market. This becomes necessary since pipe diameters are made commercially in certain discrete sizes. The problem is therefore to find a set of pipe diameters to satisfy all the constraints at the least cost. The method used to do this is a combination of steepest descent (ascent) and dynamic programming.

Since the cost function involves flow in pipes, it is necessary to calculate the flow quickly. Flow is also important in the calculation of pressure since pressure is a function of flow. To compute flow quickly the method of linear analysis which was developed for this purpose is employed.

Problem formulation - Any problem of optimization has essentially two characteristics (1) a cost function and (2) one or more constraints. For the hydraulic network these are described as follows:

(1) Cost function -- The cost function used for network optimum design is divided into two parts: (a) Capital and (b) operation and maintenance costs. For the capital cost the result of the regression analysis performed by Linaweaver and Clark is used. This analysis was carried out on pipe line data for oil, gas and water pipe lines and gives a relationship between the variables, diameter, D, in inches and the capital cost, in dollars, per mile. This relationship is given by

$$\text{Capital Cost} = 1890 \; D^{1.29} \text{ per mile}$$

or

$$\text{Capital Cost} = 0.358^{1.29} \text{ per foot}$$

13

The correlation coefficient is 0.98 according to the article.

At the time of this survey the Engineering News Record Construction Cost Index (ENRI) was 877. This enables the capital cost relationship to be updated by the ratio, PRESENT ENRI/877. The procedure described is used herein for the capital cost portion of the cost analysis. Alternate schemes could be used to express the capital cost as some continuous function of pipe diameter.

The capital expenditure is usually incurred at the time of construction of the project, and is paid back over the life of the project. During that time, the value of money is determined by the rate of interest, $i\%$ per annum. In order to spread the capital cost evenly over the whole life of the project, it is necessary to multiply the initial cost by a capital recovery factor (crf) where:

$$\text{crf} = \left[\frac{i}{100}\left(1 + \frac{i}{100}\right)^{nn}\right] \bigg/ \left[\left(1 + \frac{i}{100}\right)^{nn} - 1\right]$$

where nn = life of project in years.

Thus in a system of m pipes the annual capital cost is

$$\sum_{i=1}^{m} 0.358\, L_i D_i^{1.29}\,(\text{crf})\, \frac{(\text{PRESENT ENRI})}{877}$$

where L_i = length in feet of i-th pipe.

The operation and maintenance portion of the cost function is obtained by first equating the pumping power required for each pipe to that of an equivalent number of kilowatt-hours and then multiplying the number of Kilowatt-hours by the corresponding unit cost. The power utilized in a pipe is related to the corresponding head loss. The Hazen-Williams empirical expression for head loss is used. This expression is $H_L = KQ^{1.8518}$

$$\text{where} \qquad K = \frac{4.77 L (12)^{4.87}}{C^{1.8518} D^{4.87}} \qquad\qquad (26)$$

14

and   C = roughness coefficients.

The power lost in the pipe line is related to H_L in the following way:

Power loss = $\dfrac{H_L Q \gamma}{550}$ horse power (where Q is in cfs and γ is in the specific weight of the liquid in Lb/ft^3).

Inserting the head loss equation in this expression gives:

Power loss = $\dfrac{KQ^{2.8518}}{550}$ (62.4) horse power (assuming water is the liquid).

The final annual cost due to maintenance and operation in a pipe can be expressed as:

$$KQ^{2.8518} \dfrac{(62.4)}{550} (0.746)(365 \times 24)c \text{ per year}$$

where c = unit cost of electricity in $ per KwH. Thus the total cost function (R) is

$$R = \sum_{i-1}^{m} 0.358 \, L_i D_i^{1.29}(crf)(\text{Present ENRI}/877) \, +$$

$$\sum_{i=1}^{m} K_i Q_i^{2.8518} \dfrac{(62.4)}{550} (0.746)(365 \times 24)c = R \qquad (27)$$

The factor (365 x 24) assumes that the system is in operation for the whole year. If this is not the case, then this factor can be replaced by the anticipated number of hours in the year that the system will be in operation.

(2) Constraints -- The primary constraint is one which requires the flows to obey basic hydraulic relationships involving continuity of flow at junctions and head losses in the individual loops.

Another type of constraint, dealing with pressures, assumes alternate forms. The system must be designed for a maximum

value of pressure which must not be exceeded.

There also may be a minimum pressure required for each junction which is necessary to maintain acceptable system performance. An example of such a minimum is that required by fire-fighting activities for which standards have been developed by National Board of Fire Underwriters (NBFU)(33). It is assumed that though there are variations in minimum pressure, there is none in the maximum.

Acceptable pipe diameters are also constrained. NBFU recommends that the diameter of street mains should not be less than 6-inches. This is again a fire-protection provision where the primary consideration is to obtain an acceptable quantity of water. In addition, diameters available on the commercial market are discrete and not continuous. A 6-inch pipe may be available, while 6.25-inch usually is not. It is therefore necessary that the sizes selected for design must be commercially available.

(3) The mathematical model -- The problem is summarized as follows:

The cost function to be minimized is

$$R = \sum_{i=1}^{m} 0.358 \, L_i D_i^{1.29} \frac{(\text{Present ENRI})}{877}(\text{crf}) +$$

$$\sum_{i=1}^{m} K_i Q_i^{2.8518} \frac{(62.4)}{550} (0.746)(365 \times 24)c$$

Subject to the following constraints:

(i) pressure and flow obey basic hydraulic relationships

(ii) $p \leq \text{ABMAX}$ where p is pressure and ABMAX is absolute maximum pressure

(iii) $p \geq \text{ABMIN}$ where ABMIN is the absolute minimum pressure allowed

(iv) $p_A \geq P_A$ where p_A is pressure at junction A and P_A is the minimum allowed at junction A

16

(v) $D \geq DMIN$ where DIMIN is the absolute minimum size diameter allowed

(vi) $D \in (d_1, d_2, d_3, \ldots, d_n)$ where d_1, d_2, $\ldots$, d_n are the available commercial size diameters.

Method of Solution   The law of continuity is expressed in terms of flow ($\Sigma Q_i = 0$) while the "head loss" or loop equations ($\Sigma K_i Q_i^n = 0$) are functions of flow and pipe properties. The Hazen-Williams line loss expression is used herein. Thus from (26)

$$K = \frac{4.77L(12)^{4.87}}{C^{0.8518}D^{4.87}}$$

which is referred to as the loss coefficient. Flow is given by

$$Q = Av \qquad\qquad (28)$$

where A is the cross-sectional area and v the velocity

It can be demonstrated that the differential dQ is given by

$$dQ = \frac{2Q}{D} \, dD + A \, dv \qquad\qquad (29)$$

To preserve continuity, the algebraic sum of the changes in flow at any junction must be zero.

Thus at any junction

$$\Sigma \, dQ_i = 0 \qquad\qquad (30)$$

and from eqn. (29)

$$\Sigma A_i dv_i = -2\Sigma \frac{Q_i}{D_i} \, dD_i \qquad\qquad (31)$$

It is clear that by considering changes in flow at many junctions, equations involving $A_i dv_i$ and dD_i can be obtained. It also follows from the continuity equations that one of these equations would be redundant. The system equations can be expressed in the following form:

17

$$\begin{bmatrix} \text{Matrix coefficients} \\ \text{are} \quad 1, \ 0, \ -1 \end{bmatrix} \begin{bmatrix} A_1 dv_1 \\ A_2 dv_2 \\ A_3 dv_3 \\ \cdot \\ \cdot \\ \cdot \\ A_m dv_m \end{bmatrix} = \begin{bmatrix} \text{Matrix} \\ \text{coefficients} \\ \text{are} \quad -\dfrac{2Q}{D} \\ \text{or } 0 \end{bmatrix} \begin{bmatrix} dD_1 \\ dD_2 \\ dD_3 \\ \cdot \\ \cdot \\ \cdot \\ dD_m \end{bmatrix} \tag{32}$$

By considering first order derivatives of the loop equations, another set of equations involving $A_i dv_i$ and dD_i can be obtained as follows:

$$H_L = KQ^{1.8518}$$

is differentiated to obtain

$$dH_L = Q^{1.8518} \frac{dK}{dD} dD + 1.8518 \, KQ^{0.8518} \frac{\delta Q}{\delta D} dD$$

$$+ \ 1.8518 \, KQ^{0.8518} \frac{\delta Q}{\delta v} dv \tag{33}$$

From (29) $\dfrac{\delta Q}{\delta D} = \dfrac{2Q}{D}$ and $\dfrac{\delta Q}{\delta v} = A$

By substituting in (35)

$$dH_L = Q^{1.8518} \frac{dK}{dD} dD + 1.8518 \, KQ^{0.8518} \frac{2Q}{D} dD$$

$$+ \ 1.8518 \, KQ^{0.8518} A dv \tag{34}$$

From (26) $\qquad K = \dfrac{4.77 L (12)^{4.87}}{C^{1.8518} D^{4.87}}$

By differentiation

$$\frac{dK}{dD} = -\frac{4.87}{D^{5.87}} \frac{4.77 L (12)^{4.87}}{C^{1.8518}} = -4.87 \frac{K}{D} \tag{35}$$

18

Substitute (35) for $\frac{dK}{dD}$ in (34), then

$$dH_L = -4.87 \frac{KQ^{1.8518}}{D} dD + 3.7036 \frac{KQ}{D} dD$$

$$+ 1.8518 KQ^{0.8518} A dv \qquad (36)$$

Simplifying and combining terms give

$$dH_L = -1.1664 \frac{KQ^{1.8518}}{D} dD + 1.8518 KQ^{0.8518} A dv \qquad (37)$$

These equations of the form $\Sigma dH_L = 0$ can be transformed into

$$1.8518 \sum KQ^{0.8518}(A dv) = 1.1664 \sum \frac{K}{D} Q^{0.8518} dD \qquad (38)$$

or in matrix form:

$$
\begin{bmatrix}
\text{Matrix Coefficients} \\
\text{are either} \\
\pm 1.8518\, KQ^{0.8518} \\
\text{or } 0
\end{bmatrix}
\begin{bmatrix}
A_1 dv_1 \\
A_2 dv_2 \\
A_3 dv_3 \\
\cdot \\
\cdot \\
\cdot \\
A_m dv_m
\end{bmatrix}
=
\begin{bmatrix}
\text{Matrix coefficients} \\
\text{are either} \\
\pm 1.1664\, \frac{KQ^{1.8518}}{D} \\
\text{or } 0
\end{bmatrix}
\begin{bmatrix}
dD_1 \\
dD_2 \\
dD_3 \\
\cdot \\
\cdot \\
dD_m
\end{bmatrix}
$$

$$(39)$$

By combining all independent equations derived from the basic hydraulic relations, (32) and (39), the following simultaneous equations are obtained.

$$
\begin{bmatrix}
\begin{array}{c}
\text{Matrix Coefficients} \\
\text{are } \pm 1 \text{ or } 0 \\[2mm]
\hline
\text{Matrix Coefficients} \\
\text{are} \\
\pm 1.8518\, KQ^{0.8518} \\
\text{or } 0
\end{array}
\end{bmatrix}
\begin{bmatrix}
A_1 dv_1 \\
A_2 dv_2 \\
A_3 dv_3 \\
\vdots \\
A_m dv_m
\end{bmatrix}
=
\begin{bmatrix}
\begin{array}{c}
\text{Matrix} \\
\text{coefficients are} \\
-\frac{2Q}{D} \text{ or } 0 \\[2mm]
\hline
\text{Matrix} \\
\text{coefficients are} \\
\pm 1.1664\, \frac{KQ^{1.8518}}{D} \\
\text{or } 0
\end{array}
\end{bmatrix}
\begin{bmatrix}
dD_1 \\
dD_2 \\
dD_3 \\
\vdots \\
dD_m
\end{bmatrix}
$$

(The left bracket is labeled on its side: CONTINUITY EQUATIONS / HEAD LOSS EQUATIONS)

$$(40)$$

It is observed that, if m is the number of pipes in the network then the matrices on both sides of equations (40) are of order $m \times m$.

The cost function, as given by eqn. (27) is expressed in terms of the diameter, D, flow Q, and Hazen-Williams' loss coefficients. Q is a function of D and v, K is a function of D. Thus the cost function can be expressed in terms of D and v and the first derivative of the cost function can be put in a form similar to (40).

The cost function previously defined can be written as

$$R = \Sigma a_i D_i^{1.29} + \Sigma b_i K_i Q_i^{2.8518}$$

where a_i and b_i are constants. Differentiation of R with respect to D_i's and v_i's gives

$$dR = \left(1.29 \, \Sigma a_i D_i^{0.29} + \Sigma b_i \, \frac{\partial K_i}{\partial D_i} \, Q_i^{2.8518} \right) dD_i$$

$$+ \, 2.8518 \, \Sigma b_i K_i Q_i^{1.8518} \, \frac{\partial Q_i}{\partial D_i} \, dD_i$$

$$+ \, 2.8518 \, \Sigma b_i K_i Q_i^{1.8518} \, \frac{\partial Q_i}{\partial v_i} \, dv_i$$

Since $\dfrac{\partial K_i}{\partial D_i} = -4.87 \, \dfrac{K_i}{D_i}$ (from 35) and from (29) $\dfrac{\partial Q_i}{\partial D_i} = \dfrac{2Q_i}{D_i}$ and $\dfrac{\partial Q_i}{\partial v_i} = A_i$ the derivative becomes

$$dR = 1.29 \, \Sigma a_i D_i^{0.29} dD_i + 0.8336 \, \Sigma b_i \, \frac{K_i Q_i^{2.8518}}{D_i} \, dD_i$$

$$+ \, 2.8518 \, \Sigma b_i K_i Q_i^{-.8518} (A_i dv_i) \tag{41}$$

This can be expressed in matrix form as follows:

20

$$
\begin{bmatrix} 1 & e_1 & e_2 & e_3 \cdots e_m \end{bmatrix}
\begin{bmatrix} dR \\ A_1 dv_1 \\ A_2 dv_2 \\ \cdot \\ \cdot \\ \cdot \\ A_m dv_m \end{bmatrix}
= \begin{bmatrix} f_1 & f_2 & f_3 \cdots f_m \end{bmatrix}
\begin{bmatrix} dD_1 \\ dD_2 \\ dD_3 \\ \cdot \\ \cdot \\ \cdot \\ dD_m \end{bmatrix}
\tag{42}
$$

where e_1, e_2, $e_3, \ldots, e_m$ are coefficients of $A_1 dv_1$, $A_2 dv_2$, $A_3 dv_3, \ldots, A_m dv_m$ and f_1, f_2, $f_3, \ldots, f_m$ are coefficients of dD_1, dD_2, $dD_3, \ldots, dD_m$.

By combining (40) and (42) the left hand side becomes a matrix of order $(m+1) \times (m+1)$ and the right hand one of $(m+1) \times m$. It is also observed that "dR" only occurs once and that it is on the left hand side. Thus one of the columns on this side would have coefficients of

$$1, \; 0, \; 0, \; 0, \; \ldots 0.$$

Thus the following sets of equations are obtained

$$
\begin{array}{c}
\text{CONTINUITY EQUATIONS} \\
\text{LOOP EQUATIONS}
\end{array}
\left[\begin{array}{ccccc}
1 & e_1 & e_2 \cdots e_m & & dR \\
0 & \multicolumn{2}{c}{\text{Matrix coefficients}} & & A_1 dv_1 \\
 & \multicolumn{2}{c}{\text{are } \pm 1 \text{ or } 0} & & \\
0 & & & & A_2 dv_2 \\
 & & & & A_3 dv_3 \\
\hline
 & \multicolumn{2}{l}{\text{Matrix coefficients}} & & \cdot \\
 & \multicolumn{2}{l}{\text{are}} & & \cdot \\
 & \multicolumn{2}{l}{\pm\, 1.8518 KQ^{0.8518}} & & \cdot \\
 & \multicolumn{2}{l}{\text{or } 0} & & \cdot \\
0 & & & & A_m dv_m
\end{array}\right]
\begin{array}{c}
(m+1)(m+1)
\end{array}
=
\left[\begin{array}{c}
f_1 \; f_2 \; f_3 \cdots f_m \\
\text{Matrix} \\
\text{coefficients} \\
\text{are} \\
-\dfrac{2Q}{D} \text{ or } 0 \\
\hline
\text{Matrix} \\
\text{coefficients} \\
\text{are} \\
\pm\, 1.1664 \dfrac{KQ^{1.8518}}{D} \\
\text{or } 0
\end{array}\right]
\begin{array}{c}
(m+1)(m)
\end{array}
\begin{bmatrix} dD_1 \\ dD_2 \\ dD_3 \\ \cdot \\ \cdot \\ \cdot \\ dD_m \end{bmatrix}
$$

(43)

21

Before (43) can be used it is necessary to have values of Q_i and D_i. Values of D are assumed and values of Q are calculated. The pressure at the junctions are now computed and, if the pressure constraints are broken, new values of D are assumed and the process repeated until these constraints are satisfied.

In this initial stage, if new values of D are required it has been found convenient to add or subtract a constant increment to each of the diameters. It is usual to add the increments, but if the pressure at any point is greater than ABMAX the increment is subtracted.

After values of D and Q are obtained which are feasible solutions the partial derivatives with respect to diameters and velocities are computed and the matrix coefficients for (43) are not known. From these equations, by manipulating the matrices the following is obtained:

$$
\begin{bmatrix}
dR \\
A_1 dv_1 \\
A_2 dv_2 \\
A_3 dv_3 \\
\cdot \\
\cdot \\
\cdot \\
\cdot \\
A_m dv_m
\end{bmatrix}
=
\begin{bmatrix}
(m+1) \times m \\
\text{matrix} \\
 \\
\end{bmatrix}
\begin{bmatrix}
dD_1 \\
dD_2 \\
dD_3 \\
 \\
 \\
dD_m
\end{bmatrix}
\quad \text{------------} \quad (44)
$$

Since

$$dR = g_1 dD_1 + g_2 dD_2 + g_3 dD_3 + \ldots + g_m dD_m$$

(where g_1, g_2, ..., g_m are coefficients of the first row of matrix in (44), then

$$\frac{dD_1}{g_1} = \frac{dD_2}{g_2} = \frac{dD_3}{g_3} = \ldots = \frac{dD_m}{g_m} = \rho \qquad (45)$$

To obtain an optimal value of ρ, using the method of steepest ascent, it is necessary to solve the equation $\dfrac{dR}{d\rho} = 0$. This is a complex equation and calculating the value of ρ to satisfy it is very cumbersome and at best approximate. The value assigned to ρ is a critical factor in using the steepest ascent (descent) method. Too big a value of ρ oversteps the optimum, too small a value requires too many unnecessary calculations. This makes it difficult to find the optimum value of ρ and thus a form of direct search technique was developed to determine this value.

Of the coefficients, g_1, g_2, g_3, ..., g_m the one with the biggest absolute value is chosen, say g_k, and then dD_k corresponding to g_k is given a value. The ratio dD_k/g_k is now known and from (45) all values of dD_1, dD_2, ..., dD_m are computed. These increments, dD_1, dD_2, dD_3, ..., dD_m may be either positive or negative. If g_k is negative these increments are first of all algebraically added and if g_k is positive they are subtracted.

If there is a reduction in cost and no constraints are broken, then the process of algebraically adding or subtracting can be continued, or new partial derivatives computed and the entire procedure so far repeated. If there is no reduction in cost within the specified constraints then the increments are halved and the process of algebraically adding or subtracting is continued. If again there is no reduction, the procedure of halving increments and algebraically adding or subtracting is continued until there is such reduction or the tolerance level is reached. At this stage new matrix coefficients are computed and the whole process is repeated. Thus, for an initial increment size of two inches and tolerance level of half inch, the possible increments likely to be tried after the first are one inch and half-inch.

The process eventually converges to a situation where changes in cost are negligible. At this stage most of the diameters do not satisfy constraint number (vi) that is D_i (d_1, d_2, d_3, ..., d_m) and the final step in the procedure is to impose this condition.

If D_i is the diameter corresponding to g_i where $\quad g_i \quad >$

any other g and D_i lies between d_p and d_q -- where d_p and d_q are members of the class $(d_1, d_2, d_3, \ldots, d_m)$ and D_i is not -- then D_i is increased or decreased to d_p or d_q by making $dD_i = d_p - D_i$ or $dD_i = d_q - D_i$. The other dD's are computed and all the diameters are altered proportionately. The cost is computed for both cases when $dD_i = d_p - D_i$ and $dD_i = d_q - D_i$. The cost which is cheaper is noted and the corresponding diameters are chosen.

From now on the diameter for the i-th pipe is fixed and $dD_i = 0$. The derivative of cost function with respect to D_i will not be a term of equation (40), also any derivative with respect to D_i that is involved in the matrices of (40). The process is repeated until all the diameters are now in an acceptable set.

If, in making incremental changes in diameters, constraint (v) is broken, $(D \geq DIMIN)$ and if the diameter is not that of the pipe with the largest absolute partial derivative of cost function, the diameter of the pipe breaking this constraint will be assigned the value of DIMIN. The diameter increments for the other pipes will be unaffected. If the pipe breaking the diameter constraint is the one with the largest absolute partial derivative of cost function, then not only is this diameter assigned the value DIMIN, but the changes in the other diameters are proportionately adjusted. If a pressure constraint is broken the incremental changes are reduced proportionately until the pressure constraint is satisfied.

The method just outlined above is dependent on the shape of the cost function to obtain the optimum. If the function is convex then the global optimum is obtained. The pressure constraints, especially (iv) $(p_A \geq P_A)$ may have the effect of rendering the function non-convex, i. e., a "hole" in the feasible region. In this case it is advisable to ignore the pressure constraint in the initial stages of the process.

Computer Program -   Using the method just described a computer

24

FLOWCHART FOR OPTIMUM DESIGN
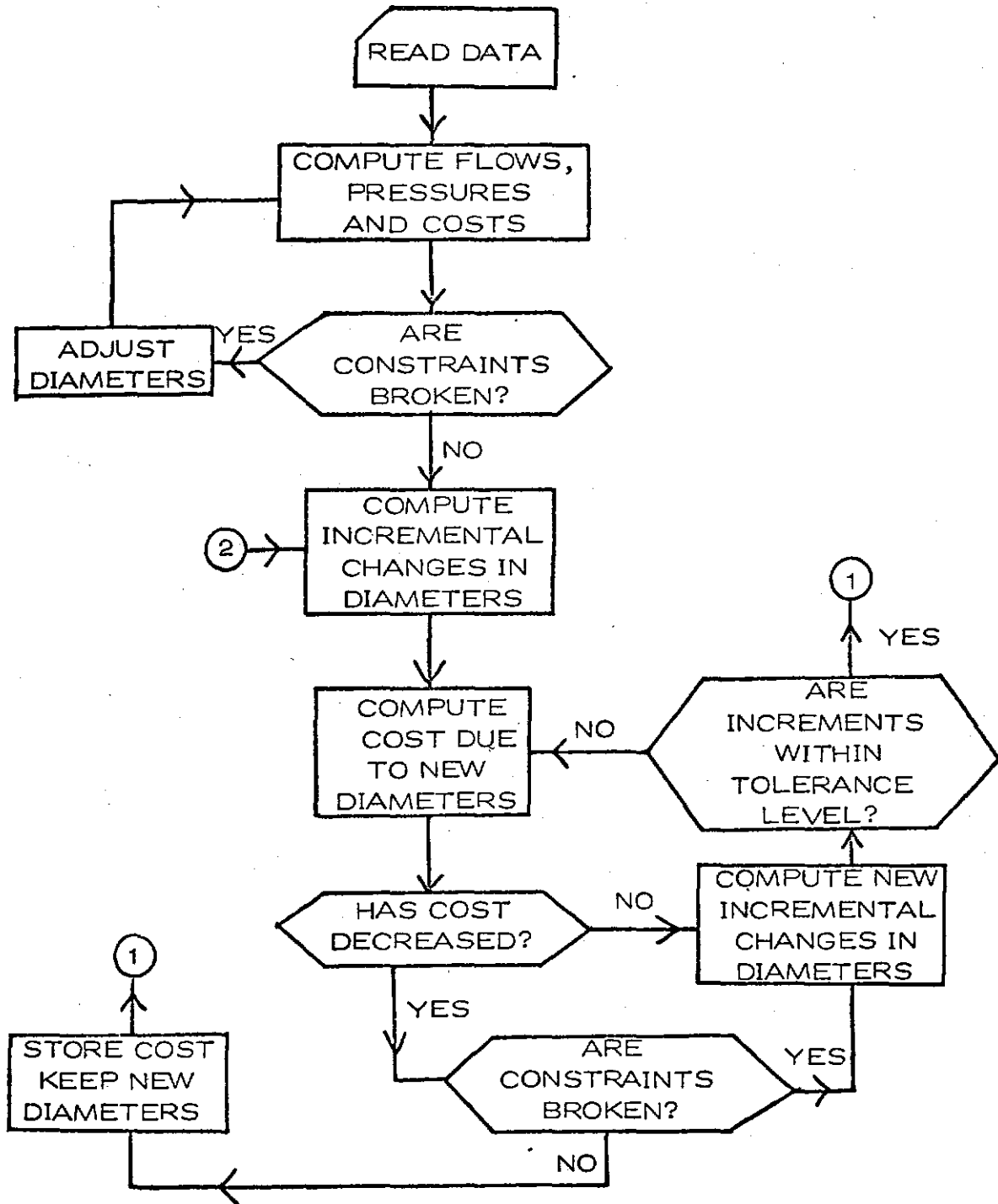
FIGURE 1A

FLOWCHART FOR OPTIMUM DESIGN
(CONT'D)

```
                    ( 1 )
                      |
                      v
              /────────────────\
             /     IS COST       \
            /    REDUCTION        \      NO
            \      WITHIN         /─────────────────►( 2 )
             \   TOLERANCE?      /                     |
              \────────────────/                      |
                      |                               |
                     YES                              |
                      |                               |
                      v                               |
          ┌──────────────────────┐                    |
          │  ADJUST DIAMETERS     │                   |
          │  AND FIX DIAMETER     │                   |
          │  WITH LARGEST         │                   |
          │  EFFECT ON COST       │                   |
          └──────────────────────┘                    |
                      |                               |
                      v                               |
              /────────────────\                      |
             /    ARE ALL        \       NO           |
            <    DIAMETERS        >──────────────────┘
             \    FIXED?         /
              \────────────────/
                      |
                     YES
                      |
                      v
             ┌────────────────┐
             │     PRINT      │
             │    RESULTS     │
             └────────────────┘
                      |
                      v
                 (  END  )
```
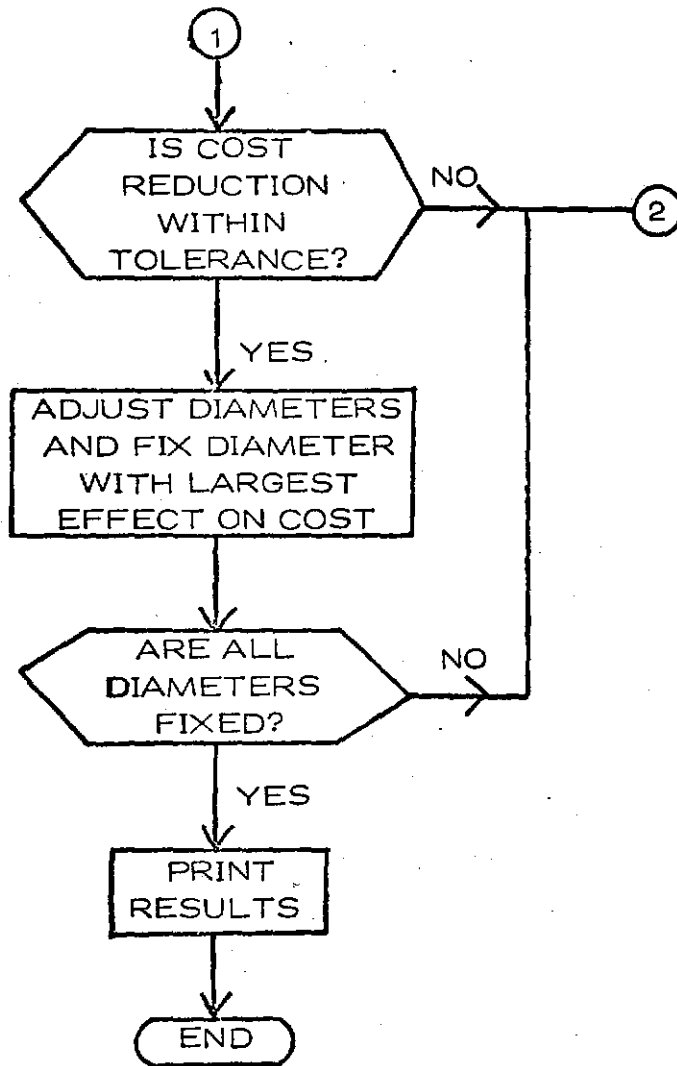
FIGURE 1B

program was written for the optimum design of closed pipe networks. A flowchart depicting the logic of this program is given in Figure 1. A listing of the program is presented in Appendix I. The program is fairly involved and the reader is referred to Reference 5 for the details.

Example - A pipe network of nineteen pipes was analyzed by this program to determine the minimum cost design. Figure 2 shows the geometry of the network. The computer output for this example is presented on the following pages to illustrate the type of information output by the computer. For the given cost information and constraints it would have been highly unlikely that this configuration could have been determined without such an aid as this program. At the present time additional efforts are being made to refine and improve this computer program. When this is completed the program will be made available to potential users.
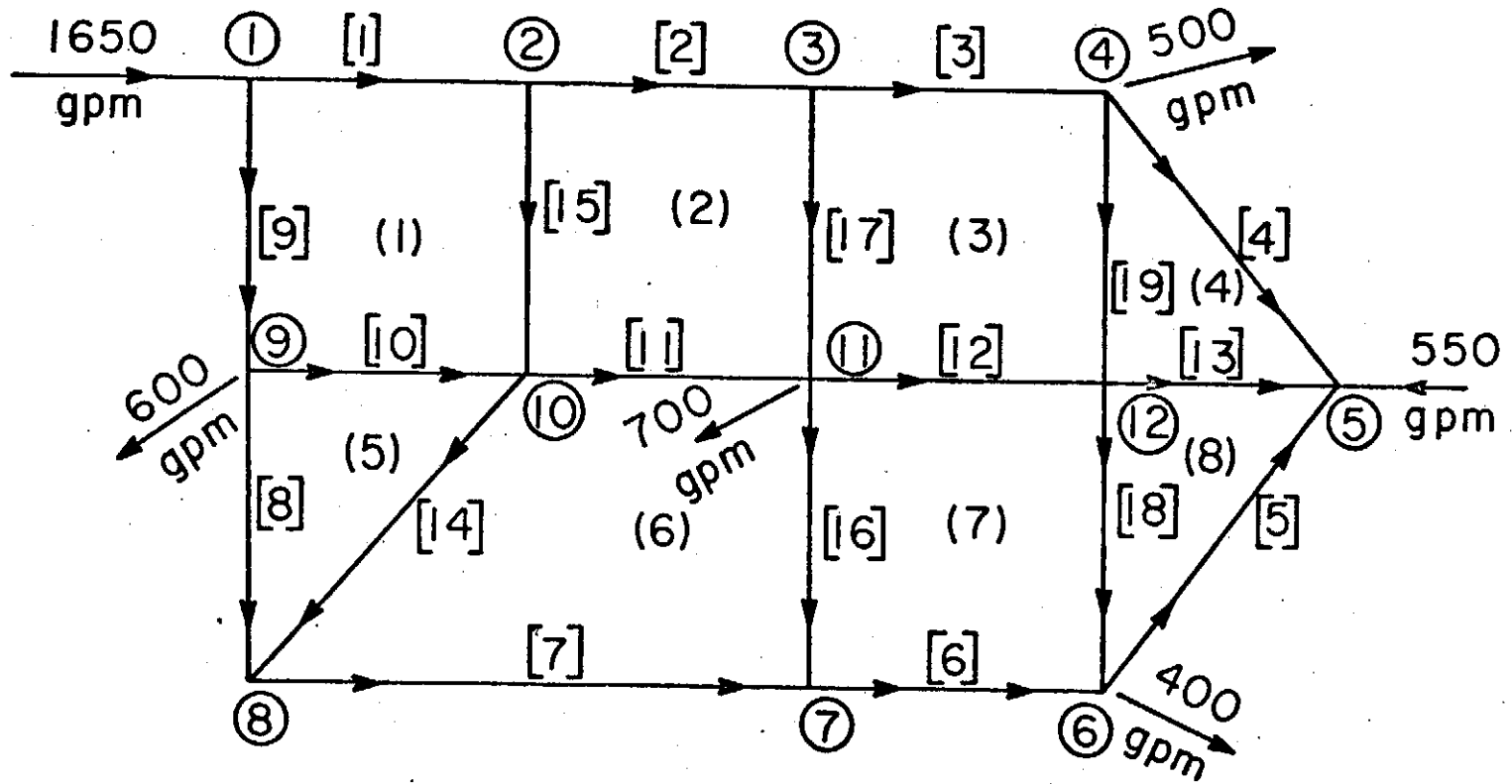
Figure 2 Hydraulic Network for Example
(to convert flowrate to m³/s multiply gpm by
6.309 × 10⁻⁵)

Pipe Number	Diameter (in.)	Length (ft.)	Roughness
1	12.0	1500	130
2	8.0	1000	130
3	8.0	1200	120
4	8.0	2000	120
5	8.0	2800	120
6	8.0	1100	120
7	8.0	1000	120
8	8.0	2500	120
9	8.0	800	100
10	6.0	1300	100
11	6.0	1000	100
12	10.0	1100	130
13	10.0	1000	130
14	6.0	1800	120
15	6.0	1100	120
16	6.0	1800	120
17	10.0	1200	130
18	6.0	1800	120
19	6.0	1300	120

TABLE I   PIPELINE DATA FOR EXAMPLE

COMPUTER OUTPUT FOR EXAMPLE

MAXIMUM PRESSURE = 150.000 LBS PER SQ IN

MINIMUM PRESSURE=   30.000 LBS PER SQ IN

MINIMUM PRESSURE ALLOWED AT JUNCTION 9 IS   50.000 LBS PER SQ IN

SMALLEST DIAMETER ALLOWED= 6.000 INCHES

GREATEST INCREMENTAL CHANGE IN DIAMETER= 6.000 INCHES

WHEN THE LARGEST INCREMENTAL CHANGE IN DIAMETERS IS LESS THAN OR EQUAL TO   0.750 INCHES
                                        SUCH CHANGES ARE IGNORED

TOLERANCE ON FLOW=15.0000 GPM

LIFE OF PROJECT= 50.000 YEARS

RATE OF INTEREST= 5.000 PER ANNUM

COST OF ELECTRICITY=$ 0.01 PER KILOWATT-HOUR

TOLERANCE ON MONEY=$ 10.00

BUILD-UP FACTOR= 1.000

ENGINEERING NEWS RECORD INDEX=   877.

RESULTS OF OPTIMAL TRIAL

PIPE NO	JUNCTION		LENGTH	ROUGHNESS	DIAMETER-INCHES		FLOW	PRESSURE AT JUNCTIONS	
					ORIGINAL	FINAL		BEGIN	END
1	1	2	1500.00	130.	10.00	6.00	311.71	120.000	123.172
2	2	3	1000.00	130.	14.00	6.00	390.92	123.172	117.600
3	3	4	1200.00	120.	16.00	6.00	286.12	117.600	106.317
4	4	5	2000.00	120.	18.00	18.00	-1106.09	106.317	105.871
5	5	6	2800.00	120.	20.00	20.00	-631.90	105.871	108.346
6	6	7	1100.00	120.	12.00	6.00	498.41	108.346	124.691
7	7	8	1000.00	120.	24.00	16.00	2431.60	124.691	126.298
8	8	9	2500.00	120.	20.00	20.00	2611.29	126.298	121.777
9	9	1	800.00	100.	18.00	24.00	4638.29	121.777	120.000
10	9	10	1300.00	100.	16.00	6.00	227.00	121.777	123.541
11	10	11	1000.00	100.	14.00	6.00	327.47	123.541	112.683
12	11	12	1100.00	130.	12.00	6.00	265.47	112.683	114.023
13	12	5	1000.00	130.	10.00	6.00	87.99	114.023	105.871
14	10	8	1800.00	120.	12.00	6.00	-179.69	123.541	126.298
15	2	10	1100.00	120.	14.00	6.00	-79.22	123.172	123.541
16	7	11	1800.00	120.	16.00	12.00	-1933.19	124.691	112.683

17	3	11	1200.00	130.	18.00	6.00			
18	6	12	1800.00	120.	20.00	6.00	104.81	117.600	112.683
19	4	12	1300.00	120.	24.00	6.00	69.68	108.346	114.023
							-107.79	106.317	114.023

CAPITAL COST=$ 12350.43

OPTIMAL COST=$ 13970.22

CORE USAGE   OBJECT CODE=   31512 BYTES,ARRAY AREA=   119800 BYTES,TOTAL AREA AVAILABLE=   170080   BYTES

DIAGNOSTICS   NUMBER OF ERRORS=   0, NUMBER OF WARNINGS=   0, NUMBER OF EXTENSIONS=   0

COMPILE TIME=   4.45 SEC,EXECUTION TIME=   716.00 SEC,   WATFIV – VERSION 1 LEVEL 3 MARCH 1971   DATE= 72/344

CONCLUSIONS

The linear method developed as a result of this study for the hydraulic analyses of water distribution systems is proving to be a valuable aid to practicing engineers. This is supported by the large number of engineers who are presently using the program.

This method is also a useful tool in the minimum cost study of water distribution systems. The feasibility of developing a routine for minimum cost design also has been established through this study. A working program for closed loop systems has been developed.

A method for solving the hydraulics of water distribution system on a standard analog computer also has been developed. This provides the basic tool for an analog-digital model for optimum network design. However, while this appears to be a very promising technique its practicality is limited. This is because most practicing engineers do not have access to the necessary analog-digital systems.

PUBLICATIONS

1. Wood, D. J., "Analog Analysis of Water Distribution Networks,"
   Transportation Engineering Journal, Proceedings, ASCE,
   Vol. 97, No. TE2, May 1971, 281-290.

2. Wood, D. J., and Charles, C. O., "Hydraulic Network Analyses
   Using Linear Theory," Proceedings, ASCE, Vol. 98,
   No. HY7, July 1972, pp. 1157-1170.

3. Charles, C. O., "Optimum Design of Hydraulic Networks Using
   Steepest Descent (Ascent) Method and Dynamic Programming,"
   Ph. D. thesis, University of Kentucky, January 1973
   (unpublished).

4. Wood, D. J., "Users Guide for Linear Method of Analysis of
   Water Distribution Systems," Department of Civil
   Engineering, University of Kentucky, 1973 (unpublished).

5. Wood, D. J., "Users Guide for Path Method of Analysis of
   Water Distribution Systems," Department of Civil
   Engineering, University of Kentucky, 1973 (unpublished).

APPENDIX I

LISTING OF FORTRAN PROGRAM
FOR OPTIMUM DESIGN OF CLOSED LOOP SYSTEMS

```
     1           IMPLICIT REAL*8 (A-H,O-Z)
     2           INTEGER UNITS,TREE(50)
     3           REAL*8 K(50),L(50),DABS,LIFE,DELEV(50),STAN(2500),SCAN(50)
     4          ODIMENSION A(50,50),C(50,50),SAVE(2500),CHAIN(2500),
                1ROUGH(50),DPRE(50),PRESS(50),HT(50),Y(50),PECED(50),D(50),DIA(50),
                ABODE(50),DEMAND(50),SD(50),SDELD(50),SQ(50),SDPRE(50),SK(50),
                2DELD(50),Q(50),LOOP(50,50),ML(50),MN(50),JBIGIN(50),JEND(50),LZ(50
                3) ,NPIPE(50)
     5           READ (5,1000) UNITS,MXX,MMX
     6      1000 FORMAT (16I5)
     7      1700 FORMAT (//18X,'RATE OF INTEREST=',F6.3,1X,'PER ANNUM')
     8      1800 FORMAT (//18X,'WHEN THE LARGEST INCREMENTAL CHANGE IN DIAMETERS IS
                1 LESS THAN OR EQUAL TO ',1X,F6.3,1X,'INCHES'  /58X,'SUCH CHANGES
                2ARE IGNORED')
     9      1900 FORMAT ('1',18X,'MAXIMUM PRESSURE =', F9.3,1X,'LBS PER SQ IN')
    10      2010 FORMAT (//18X,'MINIMUM PRESSURE=',F9.3,1X,'LBS PER SQ IN')
    11      2020 FORMAT (//18X, 'MINIMUM PRESSURE ALLOWED AT JUNCTION',I3,1X,'IS',
                1F9.3,1X,'LBS PER SQ IN')
    12      2030 FORMAT (//18X, 'TOLERANCE ON FLOW=', F7.4,1X, 'CFS')
    13      2031 FORMAT (//18X, 'TOLERANCE ON FLOW=', F7.4,1X, 'GPM')
    14      2032 FORMAT (//18X, 'TOLERANCE ON FLOW=', F7.4,1X, 'MGD')
    15      2040 FORMAT (//18X,'TOLERANCE ON MONEY=$', F8.2,1X)
    16      2050 FORMAT (//18X,'GREATEST INCREMENTAL CHANGE IN DIAMETER=', F6.3,1X,
                1'INCHES')
    17      2060 FORMAT (//18X, 'SMALLEST DIAMETER ALLOWED=',F6.3,1X,'INCHES')
    18      2070 FORMAT (//18X, 'LIFE OF PROJECT=', F7.3,1X,'YEARS')
    19      2080 FORMAT (//18X, 'COST OF ELECTRICITY=$',F5.2,1X ,'PER KILOWATT-HOUR
                1')
    20      2090 FORMAT (//18X, 'ENGINEERING NEWS RECORD INDEX=', F7.0)
    21      1701 FORMAT (//18X, 'BUILD-UP FACTOR=', F6.3)
     C         IF UNITS=1, FLOW IS IN CFS, IF UNITS=2, FLOW IS IS GPM, IF UNITS=3 FLOW IS
     C         IN MGD.  MXX=1 INDICATES THAT UNITS OF PRESSURE AT INLET ARE LBS/IN**2,
     C         NOT 'FOOT-HEAD OF WATER'
    22           READ (5,1000) NJ,NP
    23           READ (5,6000) ABMAX,ABMIN,DIMIN,DD1,DD2,DD3,DD4
     C         NJ= NO OF JUNCTIONS,NP=NO OF PIPES
    24           READ (5,2000) (JBIGIN(I),JEND(I),D(I),ROUGH(I),L(I)), I=1,NP)
    25           TOTDEM =0.0
    26           TOTAL=0.0
    27           NN=0
    28           KXXK=0
    29      1111 FORMAT (1X,1H,10X,'O U T P U T')
    30           WRITE (6,1111)
    31           WRITE (6,1900) ABMAX
    32           WRITE (6,2010) ABMIN
    33      3000 FORMAT   (5F10.4)
    34      2000 FORMAT (2I5,F5.2,2F10.5)
    35           DO 11 I=1,NJ
    36           READ (5,3000)   PECED(I),DEMAND(I),DELEV(I),DPRE(I),HT(I)
     C         PECED(I) =1.0 INDICATES NODE IS AN INLET.  DEMAND (WHETHER INLET OR OUTLET
     C         ) IS ALWAYS POSITIVE. DELEV IS ELEVATION IN FEET. DPRE  AND  HT ARE IN LBS
     C         /IN**2 UNITS, EXCEPT WHEN PECED=1.0 AND MXX .NE. 1 - IN WHICH CASE DPRE
     C         MAY BE IN ' FOOT-HEAD OF WATER' UNITS FOR THAT PARTICULAR INLET
    37           IF (PECED(I) .NE. 1.0) TOTDEM =TOTDEM+ DEMAND(I)
    38           IF (PECED(I) .EQ. 1.0) TOTAL =TOTAL + DEMAND(I)
    39           IF (PECED(I) .NE. 1.0) DEMAND(I)=-DEMAND(I)
    40           IF (PECED(I) .EQ. 1.0) NN=NN+1
    41           IF (HT(I) .NE. 0.0) WRITE (6,2020) I,HT(I)
    42        11 CONTINUE
    43           IF (TOTDEM .EQ. TOTAL) GO TO 100
```

```
44          WRITE (6,5000) TOTAL,TOTDEM
45     5000 FORMAT(//20X,'TOTAL=',1X,F10.2,2X,'TOTDEM=',1X,F10.2,2X, 'INFLOW
            1IS DIFFERENT FROM OUTFLOW')
46          STOP
47      100 DO 12  I=1,NP
48          SD(I)=0.0
49          BODE(I)=D(I)
50          IF (D(I) .LT. DIMIN) D(I)=DIMIN
51          SCAN(I)=0.0
52          Q(I)=1.0
53          NPIPE(I)=0
54          IF (I .GE. NJ) DEMAND(I)=0.0
55          ML(I)=0
56          IF (ML(I) .EQ. 0) MM=1
57          IF (ML(I) .EQ. 0) LL=16
58      101 READ (5,1000) (LOOP(I,J),J=MM,LL)
       C     LOOP(I,J) MAY BE POSITIVE OR NEGATIVE ACCORDING TO ASSUMED DIRECTION OF
       C     FLOW. (EFFLUX FROM JUNCTION IS POSITIVE, INFLUX IS NEGATIVE).  REGARD LAST
       C     JUNCTION AS REDUNDANT AND IGNORE IT.  FROM I .EQ. NJ THRU I .EQ. NP
       C     LOOP(I,J) REFERS TO INDIVIDUAL COMPONENTS OF LOOP EQUATIONS (I.E. SUM OF
       C     HEAD LOSSES =0.0) WHILE FROM I .EQ. 1 THRU I .EQ. (NJ-1) LOOP(I,J) REFERS
       C     INDIVIDUAL COMPONENTS OF CONTINUITY EQUATIONS
59          DO 13 J=MM,LL
60          IF (LOOP (I,J) .EQ. 0) GO TO 12
61       13 ML(I)=ML(I)+1
62          MM=MM+16
63          LL=LL+16
64          GO TO 101
       C     IF J=16 OR MULTIPLE OF 16    SUPPLY A BLANK CARD
65       12 CONTINUE
66          IF (UNITS .EQ. 1) B=1.
67          IF (UNITS .EQ. 2) B=.002228
68          IF (UNITS .EQ. 3) B=1.5473
69     6000 FORMAT (7F10.4)
70          READ(5,9000) RATE,LIFE,CENTS,X,BUILD,ENR
71     9000 FORMAT (8F10.5)
72          KL=NJ-1
73          READ (5,1000) (TREE(I),I=1,KL)
74          IF (ENR .EQ. 0.0) ENR=877.
75          FAC= (RATE)*.01*(1.+(RATE)*.01)**LIFE/((1.+(RATE)*.01)**LIFE -1.)
76          IF (X .EQ. 0.0) X=1.0
77          IF (BUILD .EQ. 0.0) BUILD=1.0
78          WRITE (6,2060) DIMIN
79          WRITE (6,2050) DD1
80          WRITE (6,1800) DD2
81          IF (UNITS .EQ. 1) WRITE (6,2030) DD4
82          IF (UNITS .EQ. 2) WRITE (6,2031) DD4
83          IF (UNITS .EQ. 3) WRITE (6,2032) DD4
84          WRITE (6,2070) LIFE
85          WRITE (6,1700) RATE
86          WRITE (6,2080) CENTS
87          WRITE (6,2040) DD3
88          WRITE (6,1701) BUILD
89          WRITE (6,2090) ENR
90          N=1
91          INN=16
92      141 READ(5,1100) (DIA(I),I=N,INN)
93          N=N+16
94          INN=INN+16
95     1100 FORMAT (16F5.2)
       C     IF 16 OR A MULTIPLE OF 16,THEN A BLANK CARD MUST BE INSERTED
```

```
96          IF (DIA(INN-16) .NE. 0.0) GO TO 141
97          N1=0
98          DO 44 I=1,INN
99          IF (DIA(I) .NE. 0.0) N1=N1+1
100         IF (DIA(I) .EQ. 0.0) GO TO 223
101      44 CONTINUE
102     223 INN=N1
103         N=0
104         AID=DD1
105         LPL=0
106         CHANGE=0.0
107         EXTRA=0.0
108         CT2=0.0
109         KX=0
110         CT1=0.0
111         KK=0
112         M=1
113         LX=16
114         IF (MMX .EQ. 0) GO TO 118
115     809 READ (5,1000) (MN(I),I=M,LX)
116         LX=LX+16
117         M=M+16
118         IF ((LX-16) .LT. MMX) GO TO 809
119         DO 674 I=1,MMX
120         LM=MN(I)
121     674 NPIPE(LM)=1
122     118 DO 14 I=1,NP
123         IF (D(I) .LT. DIMIN) SCAN(I)=D(I)
124         IF (D(I) .LT. DIMIN) D(I)=DIMIN
125         K(I)=4.77*12.**4.87*L(I)*(DABS(Q(I)))**.8518/(ROUGH(I)**1.8518*
            1D(I)**4.87)
126         IF (UNITS .EQ. 2) K(I)=.000012*K(I)
127         IF (UNITS .EQ. 3) K(I)=2.24416*K(I)
128         DO 14 J=1,NP
129      14 A(I,J)=0.0
130     602 LL=NJ-1
131         DO 16 I=1,LL
132         LXK=ML(I)
133         DO 16 J=1,LXK
134         LN=LOOP(I,J)
135         LM=IABS(LN)
136         IF (LN .LT. 0) A(I,LM)=-1.0
137         IF (LN .GT. 0) A(I,LM)=1.0
138      16 CONTINUE
139     102 DO 17 I=NJ,NP
140         LXK=ML(I)
141         DO 17 J=1,LXK
142         LN=LOOP(I,J)
143         LM=IABS(LN)
144         A(I,LM)=K(LM)
145         IF (LN .LT. 1) A(I,LM) =-A(I,LM)
146      17 CONTINUE
147         DO 18 I=1,NP
148         DO 18 J=1,NP
149         IG= J+(I-1)*NP
150      18 SAVE(IG) =A(J,I)
151         CALL MINV (SAVE,NP,DDD,LZ,MN,NP*NP)
152         CALL GMPRD (SAVE,DEMAND,Y,NP,NP,1,NP*NP,NP*1,NP*1)
153         DO 19 I=1,NP
154         IF (KK .EQ. 0) Q(I)=Y(I)
155         IF (KK .NE. 0) Q(I)=(Q(I)+Y(I))/2.
```

```
156        K(I)=4.77*12.**4.87*L(I)*(DABS(Q(I)))**.8518/(ROUGH(I)**1.8518*
           1D(I)**4.87)
157        IF (UNITS .EQ. 2) K(I)=.000012*K(I)
158        IF (UNITS .EQ. 3) K(I)=2.24416*K(I)
159     19 CONTINUE
160        KK=KK+1
161        IF (KK .EQ. 1) GO TO 102
162        CT=0.0
163        DO 20 I=1,NP
164        DIFF=DABS(Q(I))-Y(I))
165        IF (KK .EQ. 5) GO TO 807
166        IF (DIFF .GT. DD4) GO TO 102
167    807 CT=CT+ENR*.358*D(I)**1.29*L(I)/(877.*BUILD)*FAC
168        CT=CT+ .746/550.*24.*365.*X*CENTS*K(I)*Q(I)**2*B*62.4
169     20 CONTINUE
170        CT3=CT
171        CT4=0.0
172    652 KK=0
173    125 KL=NJ-1
174        NL=0
175        LK=0
176        DO 21 I=1,KL
177        DO 22 J=1,NJ
178     22 A(I,J)=0.0
179        LM=IABS(TREE(I))
180        LN=JBIGIN(LM)
181        LO=JEND(LM)
182        IF (LN .EQ. LK) GO TO 113
183        IF (LO .EQ. LK) GO TO 113
184        IF (PECED(LN) .EQ. 1.0) NL=NL+1
185        IF (PECED(LN) .EQ. 1.0) LK=LN
186        IF (PECED(LO) .EQ. 1.0) NL=NL+1
187        IF (PECED(LO) .EQ. 1.0) LK=LO
188        IF (PECED(LN) .NE. 1.0 .AND. PECED(LO) .NE. 1.0) GO TO 113
189        IF ((NL+KL) .GT. NJ) GO TO 801
190        DO 23 J=1,NJ
191     23 A((KL+NL),J)=0.0
192    113 IF (PECED(LN) .EQ. 1.0) A((NL+KL),LN)=1.0
193        IF (PECED(LO) .EQ. 1.0) A((NL+KL),LO)=1.0
194        IF (PECED(LN) .EQ. 1.0 .AND. MXX .EQ. 1) PRESS(NL+KL)=DPRE(LN)*
           1144./62.4+DELEV(LN)
195        IF (PECED(LO) .EQ. 1.0 .AND. MXX .EQ. 1) PRESS(NL+KL)=DPRE(LO)*
           1144./62.4+DELEV(LO)
196        IF (PECED(LN) .EQ. 1.0 .AND. MXX .NE.1) PRESS(NL+KL)=DPRE(LN)+
           1DELEV(LN)
197        IF (PECED(LO) .EQ. 1.0 .AND. MXX .NE.1) PRESS(NL+KL)=DPRE(LO)+
           1DELEV(LO)
198        IF (PECED(LN) .NE. 1.0) A(I,LN)=1.0
199        IF (PECED(LO) .NE. 1.0) A(I,LO)=-1.0
200    801 IF ((NL+KL) .GT. NJ) A(I,LN)=1.0
201        IF ((NL+KL) .GT. NJ) A(I,LO)=-1.0
202        PRESS(I)=K(LM)*Q(LM)
203        IF (TREE(I) .LT. 0) PRESS(I)=-PRESS(I)
204        IF ((NL+KL) .GT. NJ) GO TO 21
205        IF (PECED(LN) .EQ. 1.0) PRESS(I)=PRESS(I)-PRESS(NL+KL)
206        IF (PECED(LO) .EQ. 1.0) PRESS(I)=PRESS(I)+PRESS(NL+KL)
207     21 CONTINUE
208        DO 24 I=1,NJ
209        DO 24 J=1,NJ
210        IG=J+(I-1)*NJ
211     24 SAVE(IG)=A(J,I)
```

```
212        CALL MINV (SAVE,NJ,DDD,LZ,MN,NJ*NJ)
213        CALL GMPRD (SAVE,PRESS,Y,NJ,NJ,1,NJ*NJ,NJ*1,NJ*1)
214   676 DO 25 I=1,NJ
215        DPRE(I)=(Y(I)-DELEV(I))*62.4/144.
216        IF (KX .NE. 0) GO TO 104
217        IF (DPRE(I) .GT. ABMAX) GO TO 105
218        IF (DPRE(I) .LT. ABMIN) GO TO 105
219        IF (DPRE(I) .LT. HT(I) .AND. HT(I) .NE. 0.0) GO TO 105
220        GO TO 25
221   105 IF (DD1 .LE. DD2) GO TO 673
222        DO 26 J=1,NP
223        D(J)=D(J)+DD1
224        IF (DPRE(I) .GT. ABMAX) D(J)=D(J)-2*(DD1)
225    26 CONTINUE
226        GO TO 118
227   104 IF (DPRE(I) .GT. ABMAX) GO TO 106
228        IF (DPRE(I) .LT. ABMIN) GO TO 106
229        IF (DPRE(I) .LT. HT(I) .AND. HT(I) .NE. 0.0) GO TO 106
230        GO TO 25
231   106 DIFF=DABS(D(N)-SD(N))
232        IF (DIFF .LE. .0001 .AND. CT4 .NE. 0.0) CT3=CT4
233        IF (DIFF .LE. .0001) GO TO 208
234        DIFF=DABS(DELD(N)-EXTRA)
235        IF (DIFF .LE. .0001) DELD(N)=EXTRA
236        IF (DELD(N) .NE. EXTRA) GO TO 671
237        DIFF=DABS(DELD(N)-0.0)
238        IF (DIFF .LE. .0001) GO TO 671
239   661 DO 207 J=1,NP
240        D(J)=D(J)-DELD(J)+SDELD(J)
241        Q(J)=SQ(J)
242   207 CONTINUE
243        NPIPE(N)=1
244        CHANGE=0.0
245        EXTRA=0.0
246        KX=0
247        GO TO 118
248   208 DO 209 J=1,NP
249        SD(J)=0.0
250   209 D(J)=D(J)+DELD(J)-SDELD(J)
251        NPIPE(N)=1
252        CHANGE=0.0
253        EXTRA=0.0
254        KX=0
255        GO TO 118
256   671 CT4=0.0
257        GO TO 205
258    25 CONTINUE
259        DIFF=DABS(EXTRA-0.0)
260        IF (DIFF .LE. .0001) EXTRA=0.0
261        IF (EXTRA .EQ. 0.0) GO TO 669
262        DIFF= DABS(DELD(N)-EXTRA)
263        IF (DIFF .LE. .0001) DELD(N)=EXTRA
264        IF (DELD(N) .EQ. EXTRA) NPIPE(N)=1
265   669 KX=KX+1
266        IF (CT4 .EQ. 0.0) GO TO 654
267   660 IF (CT1 .NE. 0.0 .AND. CT1 .GT. CT4) CT3=CT4
268   654 IF (CT1 .NE. 0.0 .AND. CT3 .EQ. CT4) GO TO 139
269        IF (CT1 .NE. 0.0 .AND. CT1 .LT. CT4) CT3=CT1
270   653 IF (CT1 .NE. 0.0 .AND. CT3 .EQ. CT1) GO TO 139
271        IF (CT4 .EQ. 0.0) GO TO 646
272        IF (CT3 .GT. CT4 .AND. DD1 .LE. DD2) CT3=CT4
```

```
273      646 IF (KX .EQ. 1) GO TO 139
274          IF (CT3  .GT. CT4 .AND.  CT4 .NE. 0.0) CT3=CT4
275          IF (CT3 .NE. CT4) GO TO 139
276          IF (DD1 .NE. AID) DD1=DD1/2.
277          DIFF=DABS(CT2-CT3)
278          IF (DIFF .LE. DD3 .AND. DD1 .LE. DD2) GO TO 673
279          IF (DD1 .LE. DD2) GO TO 139
280      139 NM=NP+1
281          KXKX=0
282          KX=1
283          DD1=AID
284          EXTRA=0.0
285          CHANGE=0.0
286          CT1=0.0
287          LPL=0
288          CT2=CT3
289          NNN=0
290          N=1
291          DO 34 I=1,NM
292          IF (I .EQ. NM) GO TO 666
293          SD(I)=0.0
294          IF (NPIPE(I) .EQ. 1) NNN=NNN+1
295          IF (NPIPE(N) .EQ. 1) N=N+1
296          NPP=NP-1
297          IF (NNN .GT. NPP/2) DD1=AID/2.
298          IF (DD1 .LE. DD2) DD1=AID
299          IF (NNN .EQ. NP) GO TO 137
300      666 DO 35 J=1,NM
301          A(I,J)=0.0
302          C(I,J)=0.0
303       35 CONTINUE
304       34 CONTINUE
305          A(1,1)=1.0
306          DO 32 I=1,NM
307          DO 33 J=1,NM
308          IF (I .NE. 1) GO TO 121
309          IF (J .EQ. 1) GO TO 142
310          A(I,J)=.746/550.*24.*365.*X*CENTS*2.8518*K(J-1)*Q(J-1)*B*62.4
            1*(-1.)
311          IF (J .EQ. NM) GO TO 33
312      142 C(I,J)=ENR*.358*1.29* D(J)**.29*L(J)/(877.*BUILD)*FAC+.746*24.*365
            1.*X*CENTS*.8336*Q(J)*DABS(Q(J))*K(J)*62.4/550.*B/D(J)
313          IF (NPIPE(J) .EQ. 1) C(I,J)=0.0
314          GO TO 33
315      121 IF (J .EQ. 1) GO TO 33
316          LN=IABS(LOOP((I-1),(J-1)))
317          IF (LOOP((I-1),(J-1)))120,32,122
318      120 IF (I .LE. NJ) A(I,(LN+1))=-1.0
319          IF (I .LE. NJ)  C(I,LN) =2.*Q(LN)/D(LN)
320          IF (NPIPE(LN) .EQ. 1) C(I,LN)=0.0
321          IF (I .GT. NJ) A(I,(LN+1))=-1.8518*K(LN)
322          IF (Q(LN) .LT. 0.0) A(I,(LN+1))=-A(I,(LN+1))
323          IF (I .GT. NJ)  C(I,LN)=-1.1664*K(LN)*Q(LN)/D(LN)
324          IF (NPIPE(LN) .EQ. 1) C(I,LN)=0.0
325          GO TO 33
326      122 IF  (I .LE. NJ) A(I,(LN+1))=1.0
327          IF  (I .LE. NJ)  C(I,LN)=-2.0*Q(LN)/D(LN)
328          IF (NPIPE(LN) .EQ. 1) C(I,LN)=0.0
329          IF (I .GT. NJ) A(I,(LN+1))=1.8518*K(LN)
330          IF (Q(LN) .LT. 0.0) A(I,(LN+1))=-A(I,(LN+1))
331          IF (I .GT. NJ) C(I,LN)=1.1664*K(LN)*Q(LN)/D(LN)
```

```
332          IF (NPIPE(LN) .EQ. 1) C(I,LN)=0.0
333       33 CONTINUE
334       32 CONTINUE
335          DO 59 I=1,NM
336          DO 60 J=1,NM
337          IG=J+(I-1)*NM
338          SAVE(IG)=A(J,I)
339          IF (I .EQ. NM) GO TO 60
340          IG=J+(I-1)*NM
341          CHAIN(IG)=C(J,I)
342          SCAN(I)=0.0
343       60 CONTINUE
344       59 CONTINUE
345          CALL MINV (SAVE,NM,DDD,LZ,MN,NM*NM)
346          CALL GMPRD (SAVE,CHAIN,STAN,NM,NM,NP,NM*NM,NM*NP,NM*NP)
347          N=1
348          DO 36 I=1,NP
349          IF (NPIPE(N) .EQ. 1) N=N+1
350          NT=1+(N-1)*NM
351          NK=1+(I-1)*NM
352          IF (DABS(STAN(NK)) .GT. DABS(STAN(NT)))N=I
353          NT=1+(N-1)*NM
354          C(1,I)=STAN(NK)
355       36 CONTINUE
356          SCAN(N)=0.0
357          IF (C(1,N) .GT..0 .AND. KXKX .EQ. 0) LOOK=1
358          IF (C(1,N) .LT..0 .AND. KXKX .EQ. 0) LOOK=0
359          IF (DD1 .GT. DD2) GO TO 123
360      673 DO 672 J=1,INN
361          DIFF=DABS(D(N)-DIA(J))
362          IF (DIFF .LE. .0001) D(N)=DIA(J)
363          IF (DIA(J) .LT. D(N)) GO TO 672
364          IF (KX .EQ. 0) DELD(N)=DIA(J+1)-D(N)
365          IF (D(N) .EQ. DIA(J)) NPIPE(N)=1
366          IF (D(N) .EQ. DIA(J)) GO TO 118
367          LPL=0
368          IF (D(N) .LE. DIMIN) DELD(N)=DIMIN-D(N)
369          IF (D(N) .GT. DIMIN) DELD(N)=DIA(J-1)-D(N)
370          EXTRA=DIA(J)-D(N)
371      644 CHANGE=DELD(N)
372          LOOK=3
373          IF (KX .EQ. 0) KX=1
374          GO TO 668
375      672 CONTINUE
376      123 IF (DD1 .GT. DD2) DELD(N)=DD1
377          NPP=NP-1
378      668 NNN=0
379          DIFF=DABS(CHANGE-0.0)
380          IF (DIFF .LE. .0001) CHANGE=0.0
381          DIFF=DABS(C(1,N)-0.0)
382          IF (DIFF .LE. .0001) C(1,N)=0.0
383          DO 37 I=1,NP
384          IF (NPIPE(I) .EQ. 1) NNN=NNN+1
385          IF (C(1,N) .EQ. 0.0) DD1=DD2
386          IF (C(1,N) .EQ. 0.0) GO TO 651
387          DELD(I)=DELD(N)*C(1,I)/C(1,N)
388          IF (LOOK .EQ. 1) DELD(I)=-DELD(I)
389          IF (SCAN(I) .EQ. 0.0) D(I)=D(I)+DELD(I)
390          M=0
391          IF (SCAN(I) .NE. 0.0 .AND. D(I) .EQ. DIMIN) M=I
392          IF (D(I) .EQ. DIMIN .AND. SCAN(I) .NE. 0.0) D(I)=SCAN(I)+DELD(I)
```

```
393          IF (M .EQ. I) GO TO 804
394          IF (SCAN(I) .NE. 0.0 .AND. D(I) .NE. DIMIN) D(I)=D(I)+DELD(I)
395      804 IF (D(I) .GE. DIMIN) GO TO 37
396          LPL=I
397          IF (LPL .NE. N) SCAN(I)=D(I)
398          IF (LPL .NE. N) GO TO 37
399          D(LPL)=D(LPL)-DELD(LPL)
400          DELD(LPL)=DIMIN-D(LPL)
401          DO 670 J=1,NP
402          IF (J .LT. LPL) D(J)=D(J)-DELD(J)
403          DELD(J)=DELD(LPL)*C(1,J)/C(1,LPL)
404          D(J)=D(J)+DELD(J)
405          D(LPL)=DIMIN
406      670 CONTINUE
407          IF (DELD(N) .EQ. 0.0) DD1=DD2
408          IF (DELD(N) .EQ. 0.0) GO TO 651
409          GO TO 127
410       37 CONTINUE
411          GO TO 127
412      127 KKK=0
413          DO 78 I=1,NP
414          IF (D(I) .LT. DIMIN) SCAN(I)=D(I)
415          IF (D(I) .LT. DIMIN) D(I)=DIMIN
416          K(I)=4.77*12.**4.87*L(I)*(DABS(Q(I)))**.8518/(ROUGH(I)**1.8518*
             1D(I)**4.87)
417          IF (UNITS .EQ. 2) K(I)=.000012*K(I)
418          IF (UNITS .EQ. 3) K(I)=2.24416*K(I)
419          IF (NNN .EQ. NPP .AND. I .NE. N) DELD(I)=0.0
420          DO 78 J=1,NP
421       78 A(I,J)=0.0
422          DO 79 I=1,NP
423          LXK=ML(I)
424          DO 65 J=1,LXK
425          LM=IABS(LOOP(I,J))
426          A(I,LM)=1.0
427          IF (LOOP(I,J) .LT. 0) A(I,LM)=-1.0
428       65 CONTINUE
429       79 CONTINUE
430      144 DO 82 I=NJ,NP
431          LP=ML(I)
432          DO 83 J=1,LP
433          LM=IABS(LOOP(I,J))
434          A(I,LM)=K(LM)
435          IF (LOOP(I,J) .LT. 0)A(I,LM)=-K(LM)
436       83 CONTINUE
437       82 CONTINUE
438          DO 84 I=1,NP
439          DO 84 J=1,NP
440          IG= J+(I-1)*NP
441       84 SAVE(IG)=A(J,I)
442          CALL MINV (SAVE,NP,DDD,LZ,MN,NP*NP)
443          CALL GMPRD (SAVE,DEMAND,Y,NP,NP,1,NP*NP,NP*1,NP*1)
444          DO 85 I=1,NP
445          IF (KKK .EQ. 0) Q(I)=Y(I)
446          IF (KKK .NE. 0) Q(I)=(Q(I)+Y(I))/2.
447          K(I)=4.77*12.**4.87*L(I)*(DABS(Q(I)))**.8518/(ROUGH(I)**1.8518*
             1D(I)**4.87)
448          IF (UNITS .EQ. 2) K(I)=.000012*K(I)
449          IF (UNITS .EQ. 3) K(I)=2.24416*K(I)
450       85 CONTINUE
451          KKK=KKK+1
```

```
452          IF (KKK .EQ. 1)  GO TO 144
453          IF (DD1 .LE. DD2) GO TO 713
454          IF (CT4 .EQ. 0.0) GO TO 713
455          IF (KX  .NE. 1 .AND. CT3 .GT. CT4) CT3=CT4
456      713 CT=0.0
457          DO 86 I=1,NP
458          DIFF =DABS(Q(I)-Y(I))
459          IF (KKK .EQ. 5) GO TO 808
460          IF (DIFF .GT. DD4) GO TO 144
461      808 CT=CT+ENR*.358*D(I)**1.29*L(I)/(877.*BUILD)*FAC
462          CT=CT+ .746/550.*24.*365.*X*CENTS*K(I)*Q(I)**2*8*62.4
463       86 CONTINUE
464          CT4=CT
465          IF (LOOK .EQ. 3) GO TO 667
466          IF (DD1 .GT. DD2) GO TO 212
467      667 DIFF=DABS(CHANGE-0.0)
468          IF (DIFF .LE. .0001) GO TO 212
469          DIFF=DABS(CHANGE-DELD(N))
470          IF (DIFF .LE. .0001)  CHANGE=DELD(N)
471          IF (DELD(N) .NE. CHANGE) GO TO 212
472      663 CT1=CT4
473          DO 206 I=1,NP
474          SDELD(I)=DELD(I)
475          SD(I)=D(I)
476          D(I)=D(I)-DELD(I)
477          SQ(I)=Q(I)
478          SK(I)=K(I)
479      206 CONTINUE
480          DELD(N)=EXTRA
481          LPL=0
482          GO TO 668
483      212 DIFF=DABS(EXTRA-0.0)
484          IF (DIFF .LE. .0001) EXTRA=0.0
485          IF (EXTRA .NE. 0.0 .AND. CT4 .LT. CT1) GO TO 125
486          IF (EXTRA .NE. 0.0) GO TO 655
487          IF (KX .EQ. 1 .AND. DD1 .LE. DD2) GO TO 673
488      662 DIFF=DABS(EXTRA-DELD(N))
489          IF (DIFF .LE. .0001) EXTRA=DELD(N)
490          IF (DELD(N) .EQ. EXTRA .AND. CT4 .LT. CT1)   GO TO 125
491          IF (DELD(N) .NE. EXTRA) GO TO 718
492          DIFF=DABS(DELD(N)-0.0)
493          IF (DIFF .LE. .0001) GO TO 659
494      651 IF (LOOK .NE. 3) LOOK=LOOK-1
495          IF (LOOK .LT. 0) LOOK=1
496          KXKX=KXKX+1
497          IF (KXKX .EQ. 1 .AND. CT2 .LE. CT3)  DD1=1.5*DD2
498          IF (KXKX .EQ. 1 .AND. CT2 .LE. CT3) GOTO 123
499          IF (CT1 .EQ. 0.0 .AND. CHANGE .EQ. 0.0) GO TO 673
500      655 DO 211 J=1,NP
501          D(J)=D(J)-DELD(J)+SDELD(J)
502          D(J)=SD(J)
503          Q(J)=SQ(J)
504          K(J)=SK(J)
505      211 CONTINUE
506          DIFF=DABS(EXTRA-DELD(N))
507          IF (DIFF .LE. .0001) EXTRA=DELD(N)
508          IF (DELD(N) .EQ. EXTRA) GO TO 125
509      718 IF (CT3 .GT. CT4) GO TO 125
510      205 DO 203 I=1,NP
511          IF (SCAN(I) .EQ. 0.0) D(I)=D(I)-DELD(I)
512          IF (SCAN(I) .NE. 0.0 .AND. D(I) .EQ. DIMIN) M=1
```

```
513          IF (D(I) .EQ. DIMIN .AND. SCAN(I) .NE. 0.0) D(I)=SCAN(I)-DELD(I)
514          IF (M .EQ. 1) GO TO 203
515          IF (SCAN(I) .NE. 0.0 .AND. D(I) .NE. DIMIN) D(I)=D(I)-DELD(I)
516      203 CONTINUE
517          DD1=DD1/2.
518          IF (KX .EQ. 1 .AND. DD1 .LE. DD2) GO TO 651
519          DIFF=DABS(CT2-CT3)
520          IF (DD1 .LE. DD2 .AND. DIFF .LE. DD3) GO TO 673
521          IF (DD1 .LE. DD2) GO TO 118
522      659 DIFF=DABS(EXTRA-0.0)
523          IF (DIFF .LE. .0001) EXTRA=0.0
524          IF (EXTRA .NE. 0.0) GO TO 667
525          IF (KX .EQ. 2) GO TO 139
526          GO TO 123
527      137 CT=0.0
528     1200 FORMAT (//40X,'RESULTS OF OPTIMAL TRIAL')
529          WRITE (6,1200)
530          WRITE (6,1500)
531     15000FORMAT (//,1X,'PIPE NO',5X,'JUNCTION',5X,'LENGTH',5X,'ROUGHNESS',
              14X,'DIAMETER-INCHES',7X,'FLOW',7X,'PRESSURE AT JUNCTIONS',/50X,
              2'ORIGINAL',3X,'FINAL',18X,'BEGIN',8X,'END')
532          CT=0.0
533          CP=0.0
534          DO 62 I=1,NP
535          CT=CT+ENR*.358*D(I)**1.29*L(I)/(877.*BUILD)*FAC
536          CP=CP+ENR*.358*D(I)**1.29*L(I)/(877.*BUILD)*FAC
537          CT=CT+ .746/550.*24.*365.*X*CENTS*K(I)*Q(I)**2*B*62.4
538          LN=JBIGIN(I)
539          LG=JEND(I)
540          WRITE (6,1300) I,LN,LG,L(I),ROUGH(I),BODE(I),D(I),Q(I),DPRE(LN),
              1DPRE(LG)
541       62 CONTINUE
542          WRITE (6,1600) CP
543          WRITE (6,1400) CT3
544     1600 FORMAT (//18X, 'CAPITAL COST=$',F15.2,1X)
545     1400 FORMAT (//18X, 'OPTIMAL COST=$',F15.2,1X)
546     1300 FORMAT (3X,I3,5X,I3,2X,I3,3X,F9.2,5X,F6.0,8X,F5.2,6X,F5.2,5X,F9.2,
              1 3X,F3.3,3X,F8.3)
547          RETURN
548          END
       C                                                                    MINV   10
       C        ..................................................................MINV   20
       C                                                                    MINV   30
       C            SUBROUTINE MINV                                         MINV   40
       C                                                                    MINV   50
       C            PURPOSE                                                 MINV   60
       C               INVERT A MATRIX                                      MINV   70
       C                                                                    MINV   80
       C            USAGE                                                   MINV   90
       C               CALL MINV(A,N,D,L,M)                                 MINV  100
       C                                                                    MINV  110
       C            DESCRIPTION OF PARAMETERS                               MINV  120
       C               A - INPUT MATRIX, DESTROYED IN COMPUTATION AND REPLACED BY  MINV  130
       C                   RESULTANT INVERSE.                               MINV  140
       C               N - ORDER OF MATRIX A                                MINV  150
       C               D - RESULTANT DETERMINANT                            MINV  160
       C               L - WORK VECTOR OF LENGTH N                          MINV  170
       C               M - WORK VECTOR OF LENGTH N                          MINV  180
       C                                                                    MINV  190
       C            REMARKS                                                 MINV  200
       C               MATRIX A MUST BE A GENERAL MATRIX                    MINV  210
```

```
574            KI=KI+N                                                    MINV 800
575            HOLD=-A(KI)                                                MINV 810
576            JI=KI-K+J                                                  MINV 820
577            A(KI)=A(JI)                                                MINV 830
578       30 A(JI) =HOLD                                                  MINV 840
     C                                                                    MINV 850
     C            INTERCHANGE COLUMNS                                     MINV 860
     C                                                                    MINV 870
579       35 I=M(K)                                                      MINV 880
580          IF(I-K) 45,45,38                                            MINV 890
581       38 JP=N*(I-1)                                                  MINV 900
582          DO 40 J=1,N                                                 MINV 910
583          JK=NK+J                                                     MINV 920
584          JI=JP+J                                                     MINV 930
585          HOLD=-A(JK)                                                 MINV 940
586          A(JK)=A(JI)                                                 MINV 950
587       40 A(JI) =HOLD                                                 MINV 960
     C                                                                    MINV 970
     C            DIVIDE COLUMN BY MINUS PIVOT (VALUE OF PIVOT ELEMENT IS MINV 980
     C            CONTAINED IN BIGA)                                      MINV 990
     C                                                                    MINV1000
588       45 IF(BIGA) 48,46,48                                           MINV1010
589       46 D=0.0                                                       MINV1020
590          RETURN                                                      MINV1030
591       48 DO 55 I=1,N                                                 MINV1040
592          IF(I-K) 50,55,50                                            MINV1050
593       50 IK=NK+I                                                     MINV1060
594          A(IK)=A(IK)/(-BIGA)                                         MINV1070
595       55 CONTINUE                                                    MINV1080
     C                                                                    MINV1090
     C            REDUCE MATRIX                                           MINV1100
     C                                                                    MINV1110
596          DO 66 I=1,N
597          IK=NK+I                                                     MINV1130
598          HOLD=A(IK)                                                  MINV1140
599          IJ=I-N                                                      MINV1150
600          DO 65 J=1,N                                                 MINV1160
601          IJ=IJ+N                                                     MINV1170
602          IF(I-K) 60,65,60                                            MINV1180
603       60 IF(J-K) 62,65,62                                            MINV1190
604       62 KJ=IJ-I+K                                                   MINV1200
605          A(IJ)=HOLD*A(KJ)+A(IJ)                                      MINV1210
606       65 CONTINUE                                                    MINV1220
607       66 CONTINUE
     C
     C            DIVIDE ROW BY PIVOT                                     MINV1230
     C                                                                    MINV1240
608          KJ=K-N                                                      MINV1250
609          DO 75 J=1,N                                                 MINV1260
610          KJ=KJ+N                                                     MINV1270
611          IF(J-K) 70,75,70                                            MINV1280
612       70 A(KJ)=A(KJ)/BIGA                                            MINV1290
613       75 CONTINUE                                                    MINV1300
     C                                                                    MINV1310
     C            PRODUCT OF PIVOTS                                       MINV1320
     C                                                                    MINV1330
614          D=D*BIGA                                                    MINV1340
     C                                                                    MINV1350
     C            REPLACE PIVOT BY RECIPROCAL                            MINV1360
     C                                                                    MINV1370
615          A(KK)=1.0/BIGA                                             -MINV1380
                                                                         MINV1390
```

44

```
616        80 CONTINUE                                            MINV1400
         C                                                        MINV1410
         C          FINAL ROW AND COLUMN INTERCHANGE              MINV1420
         C                                                        MINV1430
617           K=N                                                 MINV1440
618       100 K=(K-1)                                             MINV1450
619           IF(K) 150,150,105                                   MINV1460
620       105 I=L(K)                                              MINV1470
621           IF(I-K) 120,120,108                                 MINV1480
622       108 JQ=N*(K-1)                                          MINV1490
623           JR=N*(I-1)                                          MINV1500
624           DO 110 J=1,N                                        MINV1510
625           JK=JQ+J                                             MINV1520
626           HOLD=A(JK)                                          MINV1530
627           JI=JR+J                                             MINV1540
628           A(JK)=-A(JI)                                        MINV1550
629       110 A(JI) =HOLD                                         MINV1560
630       120 J=M(K)                                              MINV1570
631           IF(J-K) 100,100,125                                 MINV1580
632       125 KI=K-N                                              MINV1590
633           DO 130 I=1,N                                        MINV1600
634           KI=KI+N                                             MINV1610
635           HOLD=A(KI)                                          MINV1620
636           JI=KI-K+J                                           MINV1630
637           A(KI)=-A(JI)                                        MINV1640
638       130 A(JI) =HOLD                                         MINV1650
639           GO TO 100                                           MINV1660
640       150 RETURN                                              MINV1670
641           END                                                 MINV1680
         C                                                        GMPR  10
         C    ...................................................GMPR  20
         C                                                        GMPR  30
         C          SUBROUTINE GMPRD                              GMPR  40
         C                                                        GMPR  50
         C    PURPOSE                                             GMPR  60
         C       MULTIPLY TWO GENERAL MATRICES TO FORM A RESULTANT GENERAL  GMPR  70
         C       MATRIX                                           GMPR  80
         C                                                        GMPR  90
         C    USAGE.                                              GMPR 100
         C       CALL GMPRD(A,B,R,N,M,L)                          GMPR 110
         C                                                        GMPR 120
         C    DESCRIPTION OF PARAMETERS                           GMPR 130
         C       A - NAME OF FIRST INPUT MATRIX                   GMPR 140
         C       B - NAME OF SECOND INPUT MATRIX                  GMPR 150
         C       R - NAME OF OUTPUT MATRIX                        GMPR 160
         C       N - NUMBER OF ROWS IN A                          GMPR 170
         C       M - NUMBER OF COLUMNS IN A AND ROWS IN B         GMPR 180
         C       L - NUMBER OF COLUMNS IN B                       GMPR 190
         C                                                        GMPR 200
         C    REMARKS                                             GMPR 210
         C       ALL MATRICES MUST BE STORED AS GENERAL MATRICES  GMPR 220
         C       MATRIX R CANNOT BE IN THE SAME LOCATION AS MATRIX A  GMPR 230
         C       MATRIX R CANNOT BE IN THE SAME LOCATION AS MATRIX B  GMPR 240
         C       NUMBER OF COLUMNS OF MATRIX A MUST BE EQUAL TO NUMBER OF ROWGMPR 250
         C       OF MATRIX B                                      GMPR 260
         C                                                        GMPR 270
         C    SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED       GMPR 280
         C       NONE                                             GMPR 290
         C                                                        GMPR 300
         C    METHOD                                              GMPR 310
         C       THE M BY L MATRIX B IS PREMULTIPLIED BY THE N BY M MATRIX A GMPR 320
```

```
C             AND THE RESULT IS STORED IN THE N BY L MATRIX R.          GMPR 330
C                                                                        GMPR 340
C       ....................................................GMPR 350
C                                                                        GMPR 360

642       SUBROUTINE GMPRD (A,B,R,N,M,L,NTM,MTL,NTL)
643       DIMENSION A(NTM),B(MTL),R(NTL)
644       DOUBLE PRECISION A,B,R
C                                                                        GMPR 390
645       IR=0                                                           GMPR 390
646       IK=-M                                                          GMPR 400
647       DO 10 K=1,L                                                    GMPR 410
648       IK=IK+M                                                        GMPR 420
649       DO 10 J=1,N                                                    GMPR 430
650       IR=IR+1                                                        GMPR 440
651       JI=J-N                                                         GMPR 450
652       IB=IK                                                          GMPR 460
653       R(IR)=0                                                        GMPR 470
654       DO 10 I=1,M                                                    GMPR 480
655       JI=JI+N                                                        GMPR 490
656       IB=IB+1                                                        GMPR 500
657    10 R(IR)=R(IR)+A(JI)*B(IB)                                        GMPR 510
658       RETURN                                                         GMPR 520
659       END                                                           GMPR 530
                                                                         GMPR 540
```