DESIGN OF
SANITARY SEWER SYSTEMS
BY DYNAMIC PROGRAMMING

By

WILLIAM MEIN MAIN, B.SC. (ENG)

A Project Report

Submitted to the School of Graduate Studies

in Partial Fulfilment of the Requirements

for the Degree

Master of Engineering

McMaster University

July, 1975

DESIGN OF SANITARY SEWER SYSTEMS BY DYNAMIC PROGRAMMING

MASTER OF ENGINEERING (1975)
(Civil Engineering and Engineering Mechanics)

McMASTER UNIVERSITY
Hamilton, Ontario

TITLE : DESIGN OF SANITARY SEWER SYSTEMS BY DYNAMIC PROGRAMMING

AUTHOR : William Mein Main, B. Sc. (Eng.)
University of Aberdeen

SUPERVISOR : Professor A. A. Smith

NUMBER OF PAGES : x, 209

ABSTRACT

The project report examines the design criteria and procedures currently used in the design of sanitary sewer systems. A review of literature concentrates on procedures for the automated design of sanitary sewer systems. From the literature review, it is concluded that the cost optimisation of a variable layout together with details of sewer sizes and vertical alignment is not practical.

The major part of the project is the development of a program which employs dynamic programming in a multi-pass mode to optimise the cost of a branched sanitary sewer system of fixed layout. The program permits some user flexibility by allowing options for defining cost data and design criteria. Detailed documentation for operating the program is provided. A critical appraisal of the effectiveness of the algorithm and a comparison with other programs which use the dynamic programming technique is presented. A measure of the sensitivity of the optimal policy to changes in the cost structure is examined.

ACKNOWLEDGEMENTS

The author wishes to thank Dr. A. A. Smith for his assistance with this project.

Computer time was made available through an operating grant from The National Research Council of Canada and also through grants from the Science & Engineering Research Board of McMaster University.

The author wishes to thank Mrs. Elaine Dunkey for typing this project report.

TABLE OF CONTENTS

		PAGE
ACKNOWLEDGEMENTS		iv
LIST OF FIGURES		vii
LIST OF TABLES		x
CHAPTER 1.0	STATEMENT OF PROJECT OBJECTIVES	1
CHAPTER 2.0	SEWER DESIGN PROCEDURE AND LITERATURE REVIEW	2
	2.1 DESIGN PROCEDURE	3
	2.2 DISCUSSION OF DESIGN PROCEDURE	6
	2.3 COMPUTERIZATION OF DESIGN PROCEDURE	8
	2.4 OPTIMAL DESIGN OF SEWER SYSTEMS	9
	2.4.1 INTRODUCTION	9
	2.4.2 OPTIMIZATION OF LAYOUT	12
	2.4.3 OPTIMIZATION OF SEWER SIZES AND VERTICAL ALIGNMENT FOR GIVEN LAYOUT	17
	2.4.4 OPTIMIZATION OF LAYOUT, SEWER SIZES AND VERTICAL ALIGNMENT	33
	2.4.5 CONCLUSION	36

			PAGE
CHAPTER 3.0	DYNAMIC PROGRAMMING		38
CHAPTER 4.0	PROGRAM DESCRIPTION		49
	4.1	OBJECTIVES IN WRITING THE PROGRAM	49
	4.2	INTRODUCTION	50
	4.3	PROGRAM SEWER	55
	4.4	USER SUPPLIED COST FUNCTIONS	56
	4.5	SUBROUTINE DESIGN	57
	4.6	SUBROUTINE FLOW	71
	4.7	FUNCTION PIPE1 (KN, KS, SDIA)	72
	4.8	FUNCTION PIPE2 (KN, KS, SDIA)	76
	4.9	FUNCTION CSTPM (KN, KS, KX)	79
	4.10	FUNCTION COSTM (KKN UUCUT, DDCUT)	80
	4.11	FUNCTION LOCATE (IJ, LOWEST, INCRMT)	81
	4.12	MATHEMATICAL EQUATIONS	82
	4.13	COST DATA	91
	4.14	INPUT DATA	101
	4.15	OUTPUT DATA	108

PAGE

CHAPTER 5.0	DISCUSSION OF PROGRAM		112
	5.1	CORRECTNESS OF THE PROGRAM	112
	5.2	STORAGE REQUIREMENTS	119
	5.3	COARSE GRID APPROACH	121
	5.4	COMPARISON WITH OTHER PROGRAMS	124
CHAPTER 6.0	SENSITIVITY ANALYSIS		128
CHAPTER 7.0	SUMMARY AND RECOMMENDATIONS		134
REFERENCES			135
APPENDICES	DESCRIPTION		
1.	PROGRAM LISTING		139
2.	LIST OF PROGRAM VARIABLES		179
3.	FLOWCHARTS		194

LIST OF FIGURES

FIGURE	TITLE	PAGE
2.1	LIEBMAN'S (24) SEWERAGE NETWORK	13
2.2	BARLOW'S (4) SEWERAGE NETWORK	15
2.3	PIECEWISE LINEARIZATION OF SEPARABLE CONVEX FUNCTION	20
2.4	FISHER ET AL (15) DIAMETER - SLOPE RELATIONSHIP FOR MINIMUM VELOCITY	27
2.5	STATE SPACE FOR DP AND DDDP PROCEDURES	31
2.6	ARGAMAN ET AL (1) SEWERAGE NETWORK	34
3.1	PROFILE OF SEWER SYSTEM FOR FULL FLOW CONDITIONS AND SHOWING AVAILABLE HEAD	39
3.2	DIAGRAM ILLUSTRATING OPTIMUM SOLUTION CALCULATION	41
3.3	THREE STAGE SEWER SYSTEM	43
3.4	DIAGRAM TO ILLUSTRATE COMPUTATIONAL ADVANTAGE OF DYNAMIC PROGRAMMING OVER EXHAUSTIVE SEARCH TECHNIQUE	45
4.1	NETWORK FOR TEST PROBLEM 1	51
4.2	DIAGRAM SHOWING LOCATION OF STATE DATUMS	61
4.3	DIAGRAM SHOWING METHOD OF COMPUTATION OF STATE LIMITS	64
4.4	DIAGRAM SHOWING METHOD OF COMPUTATION IN THE SECOND PASS	66

FIGURE	TITLE	PAGE
4.5	DIAGRAM SHOWING ORDER OF STORAGE OF BEST DECISION ON SCRATCH FILE	69
4.6	DEFINITION SKETCH FOR A DROP MANHOLE	74
4.7	HYDRAULIC - ELEMENTS GRAPHS	77
4.8	PIPE COST vs CUT RELATIONSHIP	96
4.9	PIPE COST vs DIAMETER RELATIONSHIP	97
4.10	TYPICAL MANHOLE DETAILS	99
4.11	DATA CARD ARRANGEMENT	106
4.12	DATA CARD INPUT FOR TEST PROBLEM 1	107
4.13	OUTPUT LISTING PART 1 - TEST PROBLEM 1	109
4.14	OUTPUT LISTING PART 2 - TEST PROBLEM 1	110
5.1	DIAGRAM ILLUSTRATING BRANCH SORTING PROCEDURE	113
5.2	DIAGRAM SHOWING DISPLACEMENT OF THEORETICAL ENERGY LINE	117
6.1	NETWORK FOR TEST PROBLEM 2	130

LIST OF TABLES

TABLE	TITLE	PAGE
2.1	COMPUTER EXECUTION TIMES	21
2.2	COMPUTER EXECUTION TIMES	24
3.1	FEASIBLE DECISIONS	44
3.2	FEASIBLE SOLUTIONS	46
4.1	ROUGHNESS COEFFICIENTS	86
4.2	SEWER PIPE COST DATA PART 1	92
4.3	SEWER PIPE COST DATA PART 2	93
4.4	EXTRAPOLATED SEWER PIPE COST DATA	95
4.5	SHAFT COST DATA	98
5.1	COMPARISON OF VELOCITIES COMPUTED BY MANNING'S AND POMEROY'S EQUATIONS	116
5.2	PIPE DIAMETER vs COST	123
6.1	SENSITIVITY ANALYSIS	132

CHAPTER 1.0 STATEMENT OF PROJECT OBJECTIVES

Sewer design is a repetitious procedure which has traditionally been performed using nomographs and hand calculations. With the arrival of the computer, programs were developed which automated the hand calculation technique to permit the rapid design of sewer systems. Because these programs did not consider all possible combinations of sewer sizes, vertical alignment and costs for each link in the network, the results may not represent the cheapest sewer system cost. Some programs, currently being promoted, optimise single links only and produce results which are inefficient. To meet the requirement for a minimum cost solution, optimisation computer programs are being developed which consider all possible combinations of layout, sewer sizes and vertical alignment and which utilise an array of costs or cost functions to obtain the optimum sewer system cost.

This study reviews the current methods used to produce optimum sewer system designs, describes a dynamic programming computer program developed by the author for the design of gravity flow sanitary sewer systems and discusses its application.

6

CHAPTER 2.0 SEWER DESIGN PROCEDURE AND LITERATURE REVIEW

This chapter describes the general requirements for the design of a gravity flow sanitary sewer system, the problems of designing a sewer system to obtain minimum cost and the optimisation methods used to achieve the minimum cost solution.

2.1 DESIGN PROCEDURE

The design of a sanitary sewer system generally consists of the following operations:

1. assembly of the physical data of the area to be serviced,
2. selection of the sewer layout,
3. computation of sewer sizes and vertical alignment to conform with the design criteria set by the local authority,
4. computation of strength class of pipe required, and design of manholes,
5. estimation of cost.

The required physical data for the area includes a topographical map of the area, a land use plan, street layout drawing, soil and water table data, and location of the outfall.

From the physical data, the designer selects a suitable sewer layout to service the area.

The computation of the sewer sizes and their vertical alignment is then carried out in conformity with the local design standards and criteria. These will normally specify the following:

1. average flow rates for domestic, commercial and industrial areas,
2. peaking factors for the above flow rates,
3. infiltration rate,
4. method of computation of design flow rate(s) for the sewer,

5. hydraulic steady state conditions,
6. hydraulic resistance law, e.g. Manning's equation,
7. hydraulic roughness coefficients,
8. full flow or partial flow analysis,
9. maximum design capacity for a sewer,
10. criterion to prevent erosion, e.g. maximum velocity requirement,
11. criterion to prevent deposition of solids, e.g. minimum velocity requirement or tractive force theory,
12. method of computation of energy losses and vertical alignment at manholes,
13. minimum sewer size,
14. non-decreasing pipe sizes in a downstream direction,
15. maximum manhole spacing requirements,
16. types of pipe that may be used, e.g. concrete,
17. minimum cover requirements,
18. method of computation of strength class of pipes.

The objective of the design procedure is:

1. to design the system to carry the maximum flow rates without surcharging and at velocities which will not cause erosion, and to carry the minimum flows at velocities which prevent deposition of solids,
2. to minimize the sewer system cost.

The design of the sewer system usually starts at the upstream end and proceeds downstream. For each link, the designer selects that combination of sewer size and vertical alignment which he believes will give the cheapest sewer system cost.

Up to about 10 years ago, the design procedure was performed manually.

2.2 DISCUSSION OF DESIGN PROCEDURE

In the design of a sewer system there may be several possible layouts for the system. There is no formula available to advise the designer which layout will result in the cheapest construction cost. In general, he will select a layout which provides for maximum use of the minimum size sewers prior to joining them to large size sewers and which in his judgement is the best. The designer may produce a design for other layouts in order to obtain the optimum cost. However the expense of producing other designs does not encourage this approach.

Once a layout has been selected, the designer selects the pipe sizes and their vertical alignment. Usually for each link in the network, the designer will select a sewer slope and enters a nomograph with this slope to obtain a pipe size to carry this flow. Invariably, the theoretical diameter obtained will not be a commercial pipe size. Hence, the designer may choose to use the first commercial size pipe greater than the theoretical size requirement or he may choose to increase the slope thereby permitting the use of a commercial size pipe smaller than the theoretical size requirement. The cheaper solution for the link depends on the cost trade-off between pipe cost vs excavation and backfill costs. The choice of the cheapest solution for the link may not be the cheapest for the sewer system because of the vertical alignment and non-decreasing pipe size requirements. There are a large number of possible solutions for any sewer system. This is illustrated in the simple example of fig. 3.2.

In conclusion, it may be stated that the design of a sewer system is a sequential decision process which makes it impossible for a designer to consider all the possible solutions in order to obtain the optimum solution.

Present research is therefore directed to the optimum design of a sewer system. Ideally, the design method would optimize the layout and sewers simultaneously.

2.3 COMPUTERIZATION OF DESIGN PROCEDURE

The repetitious nature of the design of sewers is ideally suited to the use of an automated solution procedure by computer.

Sewer design packages have been developed which completely computerize the design of a given layout using minimum dig as the main cost criterion. The rapidity with which the calculations were performed meant that designs for other layouts could be easily obtained.

One of these packages is that of DiCicco, Soehngen and Takagi (9). The advantages of such computer programs consisted of the systemization of procedures, the rapid execution of calculations and a reduction of office costs. However, such programs may not give an optimum or cheapest solution. The sewer size and vertical alignment for each link in the sewer network is selected to conform to the design criteria and generally to provide minimum dig. Other possible solutions for the link are not considered when the downstream link is designed. The construction cost estimate is made after the design of the sewer sizes is completed. Costs, apart from the minimum dig consideration, are not an intrinsic part of the design of the sewer system.

2.4 OPTIMAL DESIGN OF SEWER SYSTEMS

2.4.1 INTRODUCTION

An optimum solution is defined as the cheapest sewer system installation cost subject to the constraints imposed on the system. It is recognised that the cost constraints, particularly the installation costs, are not accurately known until tenders for the construction of the sewer system are received. However the designer should design the sewer system for the cheapest cost using the best pre-tender cost estimates available. It is conceivable that, in the future, contractors might bid on a preliminary sewer system as a unit price contract, the final system design being determined from an optimization program using the tendered unit prices.

It is realized that a computer optimization process would not be worthwhile for the design of all sewer systems. Their usefulness is limited when one considers that the major cost of the supply and installation of a sewer is excavation and backfill costs. Walsh and Brown (39) stated that excavation and backfill costs represent about 80% of the total cost of the supply and installation of a sewer. Therefore, it is not worthwhile to use a computer optimization program to design a sewer system for a steep and uniform terrain as the sewer pipes will generally be located at minimum depth. However, in flat or undulating terrain, a computer optimization is useful.

As the cost of excavation and backfill decreases with respect to the cost of the pipe, the viability of computer optimization programs will improve. An interesting move, in this regard, is recorded by Mortenson (30) who describes a method for installing watermain without the use of shoring, thereby effecting a saving in excavation and backfill costs.

An important application of computer optimization programs is that they permit the analysis of the effect of the design criteria on the cost of the sewer system.

The objective of the optimization is to minimize the sewer system cost subject to the hydraulic and physical constraints and the cost data. For a gravity flow network, the objective function is principally the sum of the costs of:

1. pipe supply and installation, (including manholes)
2. excavation and backfill,

and can be formulated as a non-linear function.

The solution is subject to the following constraints:

1. minimum allowable diameter constraint,
2. diameter progression, i.e. non-decreasing diameters as design proceeds downstream,
3. invert progression, i.e. invert elevation of downstream sewer must not be greater than the invert elevation of the contributory upstream sewer,
4. minimum cover constraint,
5. minimum and maximum velocity constraints,
6. sewer capacity must equal or exceed the design flow rate.

The constraints may be linear or non-linear depending on the choice of variables for the objective function.

The process of optimization will be considered under the following headings:

1. Optimization of Sewer Layout,
2. Optimization of Sewer Sizes and Vertical Alignment for a given layout,
3. Optimization of Layout, Sewer Sizes and Vertical Alignment.

2.4.2 OPTIMIZATION OF LAYOUT

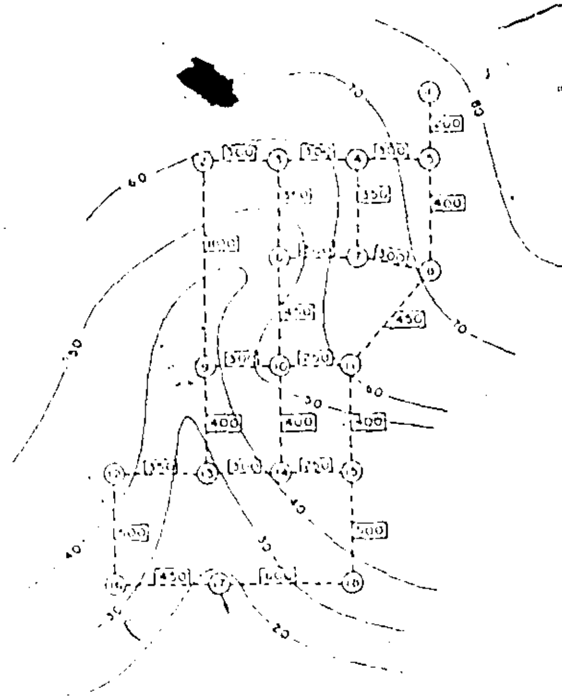
In this section, the optimization of the layout is considered with some attention to the hydraulic criteria but with no detailed consideration of the sewer sizes and their vertical alignment.

1. Trial and Error

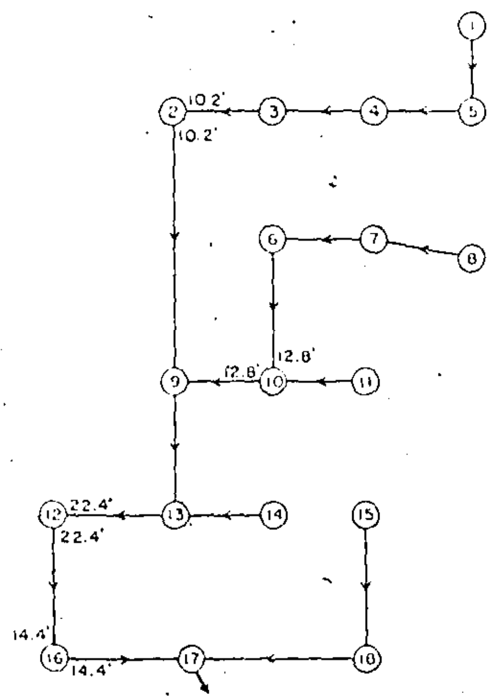
Liebman (24) suggests a trial and error method by computer for obtaining a good gravity flow sewer layout. The objective function is the sum of the pipe and the excavation costs. Pipe sizes are considered to be uniform throughout the network. The pipes are vertically aligned to suit the minimum cover, the hydraulically minimum slope and the invert progression requirement. Figure 2.1 shows the test problem from Liebman's paper. Figure 2.1a shows the layout with all the possible links; figure 2.1b shows an initial network which forms the starting layout for the computer search. The program examines the initial network to find an unused link that could be exchanged for one link in the initial layout. If this exchange produces a cost saving, the exchange is accepted. Each exchange requires that the search process must begin all over again. Execution of the search is complete when no improvement can be made.

The program was tested using the gravity flow network shown in figure 2.1. Computer execution times varied from 30 to 90s depending on the initial layout adopted.

The program does not produce an optimum solution because:



—TOPOGRAPHIC CONDITIONS FOR TRIAL NETWORK



—INITIAL LAYOUT, TRIAL A, COST = 97,[illegible] UNITS

Figure 2.1 LIEBMAN'S (24) SEWERAGE NETWORK

(i) the hydraulics of the network are not considered,

(ii) the result is dependent on the initial layout,

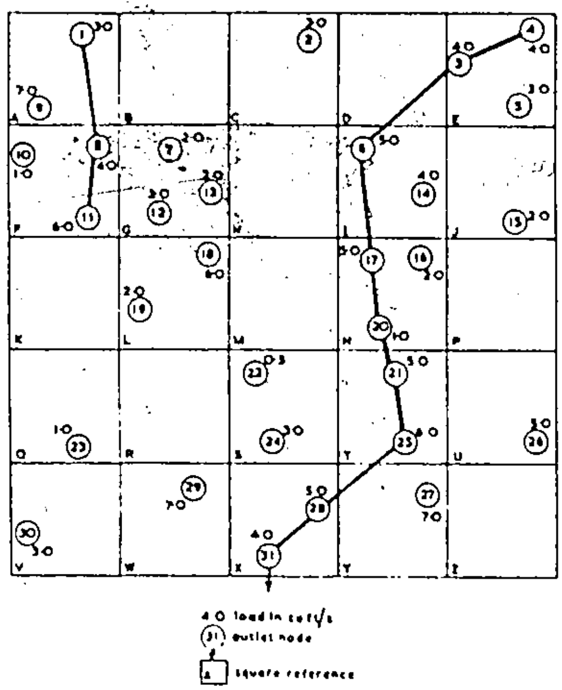
(iii) the result is dependent on the order of searching,

(iv) the method is based on single link exchange. (Multiple link exchange, i.e. exchanging two or more links simultaneously, might produce a cheaper solution).

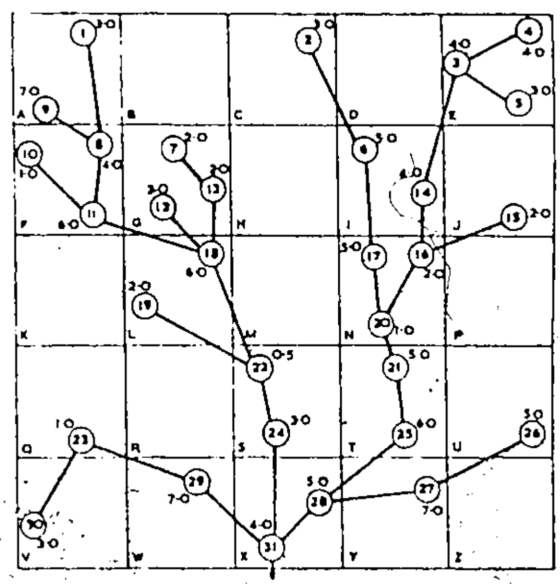
Holland (18) suggests a technique for adjusting Liebman's method to allow the flow rates to be considered. Pipes at minimum gradient could then be selected to carry these flows thereby providing a more realistic cost function.

2. Minimum Spanning Network Algorithm

Barlow's method (4) starts with the superimposing of a square grid over the sewer area and determining the sewer flow loads in each square of the grid. Starting with the most heavily loaded square adjacent to the outlet, the process proceeds upstream selecting the heaviest loaded adjacent squares until the extremity of the sewer area is reached. The outcome of this process is the identification of the most heavily loaded squares connecting the outfall to the extremity of the sewer area. A trunk sewer is then selected to traverse these squares and join the heaviest loads in these squares by the "shortest path through many points" technique. In the latter, the paths are replaced by the sewer costs. Sewer costs are computed from pipe sizes which are selected from a table of the minimum and maximum slopes and the available slope. The



Example of a major trunk and subsidiary branch



Final network showing optimal links

Figure 2.2 BARLOW'S (4) SEWERAGE NETWORK hydraulics of the sewer system is not considered.

Branch sewers are selected using the same technique.

All the remaining unconnected loads are connected to the branch network by Prim's (33) method for determining the shortest spanning tree.

Figure 2.2 taken from Barlow's article illustrates the method.

The procedure was not applied to an actual problem. The procedure did not consider the street layout.

Dajani, Gemmell and Morlok (6) suggest a similar method of obtaining a good layout by generating the shortesf path trees totally spanning a given street layout. It is based on the fact that length is the most important factor in the cost of routing flow between two points.

3. Other Methods

Templeman and Wilson (37) attempted to optimize the position of intermediate manholes within a fixed network layout. They concluded that it was not practical because manhole locations are primarily located by building location and street layout.

2.4.3 OPTIMIZATION OF SEWER SIZES AND VERTICAL ALIGNMENT FOR A GIVEN LAYOUT

The problem consists of optimizing a non-linear objective function, essentially in two variables for each link–depth of excavation and pipe diameter – subject to geometric and hydraulic constraints. The linearity or non-linearity of the constraints is dependent on the design variables used in the formulation of the objective function.

The following optimization processes have been used.

1. Linear Programming

Linear programming requires that:

(i) the objective function and the constraints be linear,

(ii) the variables in the objective function be continuous over their defined range.

Deininger (10) formulated a method of solution for a gravity flow sewer system by proposing an objective function of two variables – depth to subgrade and the pipe diameter. His objective function was:

$$\sum_{i=1}^{n} c_p d_i l_i + .5wc_e \sum_{i=1}^{n} \left[(g_i - x_i) + (g_{i+1} - y_i) \right] l_i$$

in which,

c_p = cost of pipe per foot of diameter,

d_i = diameter of pipe in reach i,

w = width of trench,

c_e = cost of excavation,

g_i = surface elevation,

x_i = invert elevation of pipe at upper end of reach i,

y_i = invert elevation of pipe at lower end of reach i,

l_i = length of reach i.

With this formulation, all the constraints except the velocity constraints are linear. From Manning's equation, $kv^2 = d^{4/3}s$

where,

k = constant

d = pipe diameter

s = slope

= $(x - y)/l$

Thus the following inequalities are obtained for each reach:

$$d_i^{4/3} s_i \geqslant kv^2 \min$$

$$d_i^{4/3} s_i \leqslant kv^2 \max$$

Deininger noted that the above inequality equations are convex and may be approximated by linear segments. With this transformation, the problem may be solved by linear programming.

Dajani, Gemmell and Morlok (6) used a convex-separable linear objective function. The objective function is based on a development by Holland (19) for the design of gravity flow sewer systems. The objective function is a non-linear function in two variables - the sum and difference of the upstream and downstream elevations of each sewer. With this formulation all the constraints are linear with respect to the two variables. Thus the problem is to minimize a non-linear objective function subject to linear constraints. The technique used to solve this is called convex-separable programming. The objective function consists of the sum of separate functions each of which involves only one variable. In addition, each of these separate functions is convex i.e. a straight line joining any two points on the function is such that the arc joining the two points lies below the straight line. Each of these separate non-linear functions can be replaced by a piecewise linear approximation as shown in figure 2.3. For each function which is linearized the following linear constraints must be added:

$$x_i = x_o + \sum_{j=1}^{n} x_j$$

$$y_i = f(x_i)$$

$$= y_o + \sum_{j=1}^{n} s_j x_j$$

$$o \leqslant x_j \leqslant \Delta x_j$$

x_j is the amount of overlap with Δx_j

This piecewise transformation of the non-linear objective function permits the solution to be obtained by a linear programming algorithm. The solution yields the optimal values of the sum and difference of the invert elevations of each sewer link. The difference of the invert elevations enables one to find the pipe diameter. Half the sum of the two variables gives the

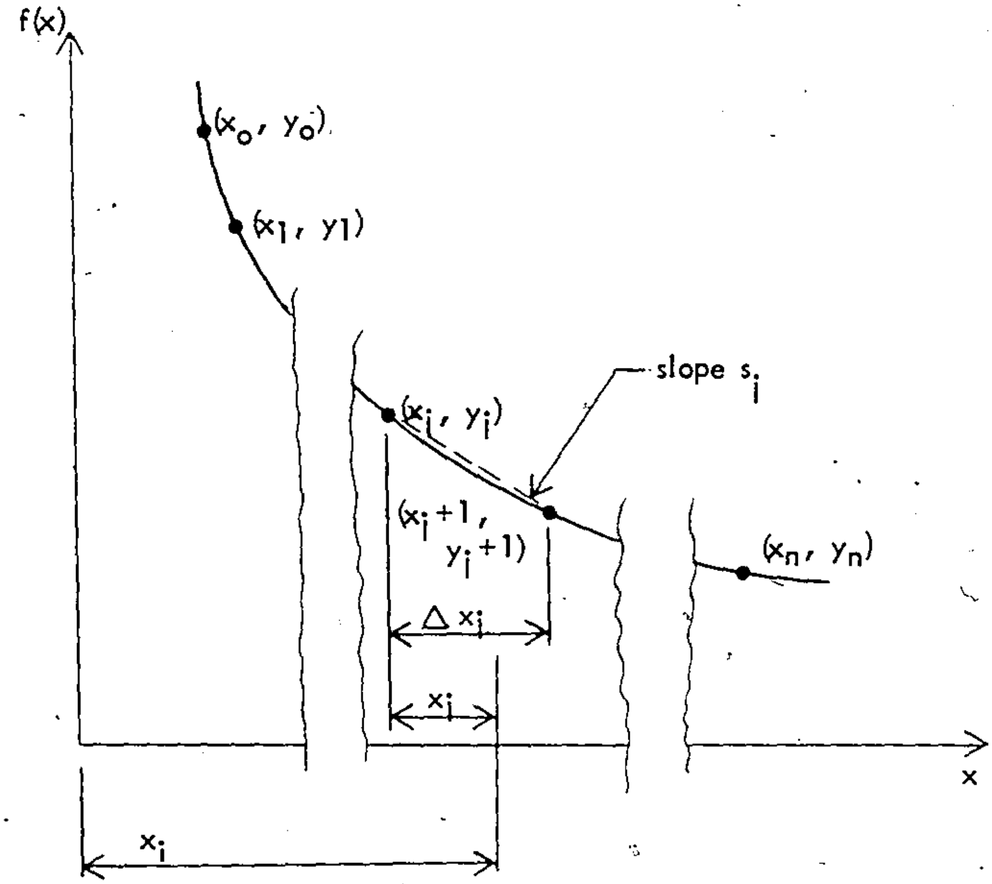


Figure 2.3 PIECEWISE LINEARIZATION OF SEPARABLE CONVEX FUNCTION

upstream invert elevation for the link. The solution gives continuous pipe sizes which must be rounded off to the nearest commercial size. Dajani, Gemmell and Morlok noted that Holland (19) had considered using a random sampling approach and had also experimented with an iterative technique applied to the optimum solution in order to obtain a solution in discrete pipe sizes.

The program was used to solve several hypothetical and actual sanitary sewer problems. The authors had reservations about the practical use of their program because of the computer execution time requirements. Each link in a network requires about 16 rows in the problem matrix. Execution time rises exponentially as the number of rows increases, and on a CDC6400 computer the following computer execution times were used.

Table 2.1 Computer Execution Times

No. of Sewers	Execution Time (s)
6	15
20	60
30	120

Dajani and Hasit (7) describe three models used to obtain the optimum cost for gravity flow, non-serial, sanitary sewer systems. The models are, respectively, based on:

(i) full flow conditions and continuous diameter,

(ii) full flow conditions and discrete diameters, and

(iii) partial flow conditions and discrete diameters.

Model (i) is similar to that program described in Dajani, Gemmell and Morlok (6). In each model, the objective function is a non-linear function in two variables - the sum and difference of the upstream and downstream invert elevations of each sewer. The constraints can be formulated in terms of these two variables and are linear with respect to them.

Model (ii) differs from model (i) in the addition of two constraints for each link in the network in order to obtain a solution in commercial pipe sizes. With this formulation, the program may be called a separable - convex mixed-integer program.

The method of obtaining the two constraints is as follows. For each feasible pipe size and given flow, there is one slope at which the pipe will flow full. The product of this slope and the sewer length is $\emptyset_{ij}$, where i = link number, j = pipe number. Hence for a given value of flow rate, a value of $\emptyset_{ij}$ will be obtained for each pipe size considered. If the difference between the upstream and downstream invert elevations for sewer i is x_{1i}, then

$$x_{1i} = \emptyset_{i1} \text{ or } \emptyset_{i2} \ldots \text{or } \emptyset_{ij} \ldots \text{or } \emptyset_{ip}$$

in which, p = number of commercial pipe sizes considered. x_{1i} must take one of the values $\emptyset_{ij}$. This requirement is expressed by the mixed integer equation:

$$x_{1i} = \emptyset_{i1}\ \delta_{i1} + \emptyset_{i2}\ \delta_{i2} + \ldots + \emptyset_{ij}\ \delta_{ij} + \ldots + \emptyset_{ip}\ \delta_{ip}$$

and;

$$\delta_{i1} + \delta_{i2} + \ldots + \delta_{ij} + \ldots + \delta_{ip} = 1$$

in which,

$$\delta_{ij} = 0 \text{ or } 1$$

With this formulation the program can be solved using IBM's Mathematical Programming System Extended with the mixed integer programming option. In model (iii) partial flow conditions are considered. The slope necessary to obtain a velocity v_s in a conduit which is equivalent to self-cleaning velocity v_f at full flow, for roughness constant with depth, is given by the following equation:

$$\frac{S_f}{S} = \left(\frac{Q}{Q_f}\frac{A_f}{A}\right)^6$$

in which,

S = slope

Q = flow rate

A = flow area

and, suffix f refers to full flow conditions. The above equation is derived from the tractive force theory.

For a given flow rate and pipe size, if 'q' values of v_s are considered, then from the equation it is possible to obtain 'q' corresponding values of S. And from the S values, q values of drop, ϕ_{ijv}, may be obtained. Suffixes i, j and v refer to the line number, pipe size number and velocity respectively. If P pipes are considered then pq values of ϕ_{ijv} will be obtained. These pq values are entered as data in the program. x_{1i} must take one of these values and this requirement is expressed by the mixed integer equation:

$$x_{1i} = \sum_{j=1}^{p} \sum_{v=1}^{q} \phi_{ijv} \, \delta_{ijv}$$

and,

$$\sum_{j=1}^{p} \sum_{v=1}^{q} \delta_{ijv} = 1$$

in which,

$$\delta_{ijv} = 0 \text{ or } 1$$

The above equations simplify the objective function because the x_{1i} term may be replaced and the piecewise linearization of the x_{1i} term is no longer required. In addition, the velocity and pipe size constraints are eliminated.

A 7-link non-serial gravity flow sanitary sewer system was solved using the 3 models on an IBM 370/165 system. The computer execution times and the optimum costs are shown below:

Table 2.2 Computer Execution Times

Model	Execution Time(s)	Cost($)
i	18	21,359
ii	40	33,471
iii	140	26,447

The table shows that model (iii) gives the cheapest realistic (model i gives continuous diameters) solution but the computer execution time is excessive. To reduce the computer execution time of the more sophisticated model (iii), Dajani and Hasit utilized the x_{1i} values obtained by running model (i). These x_{1i} values were adjusted $\pm$ 50% and pq feasible values of ϕ_{ijv} to suit the adjusted x_{1i} values were entered as data into the program. The computer execution was reduced from 140s to 60s. Thus the total execution time for the problem was 78s.

Kühner and Harrington (22) noted the following points regarding the methodology proposed by Dajani and Hasit:

(i) The test problems did not reflect a problem of real world size and computer times increased significantly with an increase in integer variables.

(ii) Approximately 50% of the computer execution time was used to prove the optimality.

(iii) The effect of the discretization of the flow velocity was not evaluated.

Kühner and Harrington suggest that in view of the uncertainty in data and computing costs, and the expensive execution time, computing strategy should aim for a good solution rather than an optimal solution.

Fisher, Karadi and McVinnie (15) tried an iterative linear integer programming method. In their formulation of the problem, the objective function is defined as the sum of the following terms:

(i) pipe supply and installation costs - a linear function with commercial pipe sizes as the variable,

(ii) excavation and backfill costs - a non-linear function with depth of excavation and sewer invert slopes as variables,

(iii) lift stations – a lump sum for each station.

To make the objective function wholly linear, a tangential approximation method was used to compute the excavation and backfill costs. The equation for these costs was of the form:

$$c_{2i} = ax_i^2 + bx_i + cs_i^2 + ds_i + ex_is_i + f$$

in which,

x_i = vertical distance from arbitrary datum to upstream invert

s_i = slope of invert for sewer i

a, b, c, d, e and f = constants determined from the trench geometry and ground elevations.

For trial values of x_i and s_i, the cost is given by:

$$c_{2i} = \left.\frac{\partial c_{2i}}{\partial x}\right|_p x_i + \left.\frac{\partial c_{2i}}{\partial s}\right|_p s_i + \text{constant}$$

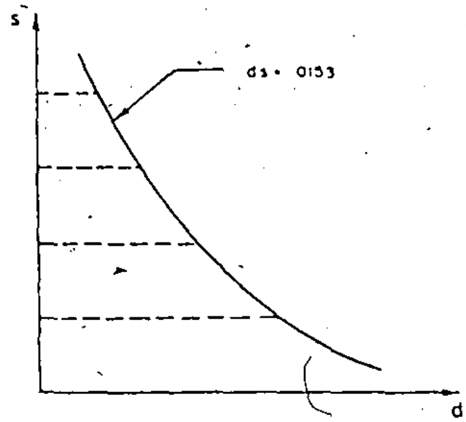
in which, P = discrete pipe size.

The minimum velocity constraint was expressed in terms of slope and diameter by the following function derived from Camp's (29) theory:

$$d_is_i \geq \text{constant}$$

The d_is_i function was approximated by a series of straight lines, the end points of which are represented by commercial pipe sizes. This is shown diagrammatically in figure 2.4.

The maximum velocity constraint was not formulated in the program. The solution was simply checked to ensure that the maximum velocity was not exceeded.



Diameter-slope relationship for minimum velocity.

Figure 2.4 FISHER et al (15) DIAMETER - SLOPE RELATIONSHIP

With this formulation the problem was completely linear and was solved iteratively using the SIMPLX method on a Univac 1108. Trial values of x_i and s_i were assumed and convergence to the allowable accuracy came within four iterations. The program was tested on a 5-link gravity flow serial trunk sewer. No execution times were provided.

They believed that their program was not a substantial improvement over the traditional methods of design because:

(i) of uncertainty of cost data,

(ii) Manning's formula does not accurately describe the hydraulic resistance,

(iii) selection of velocity criteria is not scientific.

They believe that research should be directed to the analysis of transient flow conditions.

2. Dynamic Programming

Dynamic programming (DP) is ideally suited to the design of sewer systems because:

(i) the design problem is essentially a sequential decision making process,

(ii) the formulation of dynamic programming permits the use of linear or non-linear functions or constraints,

(iii) the solution is obtained in terms of discrete diameters.

The concept of the application has been attributed to Haith (16) and the following researchers have used the technique:

(i) Argaman, Shamir and Spivak (1,2)

(ii) Templeman and Wilson (37)

(iii) Walsh and Brown (39)

(iv) Merritt and Bogan (28)

(v) Tang, Mays and Yen (26, 36, 41)

(vi) Meredith (27)

Argaman, Shamir and Spivak (1, 2) used dynamic programming for the simultaneous optimization of the layout, sewer sizes and vertical alignment. The other researchers optimized the sewer sizes and vertical alignment for a given layout.

All the researchers, except Tang, Mays and Yen (26, 36, 41), used the conventional dynamic programming code for the design of sanitary sewers. Tang, Mays and Yen used an extension of dynamic programming called discrete differential dynamic programming (DDDP) and applied this method to the design of gravity flow storm sewers.

The application of DDDP to water resources systems optimization is described by Heidari, Chow, Kotovic and Meredith (17). The DDDP process does not require searching through all the feasible states for the solution. This results in a reduction in the computer storage requirements and execution time as compared to a solution by the conventional code. The method involves selecting a trial trajectory through the feasible states and searching in a corridor around the trial trajectory to find an improved trajectory. The search starts all over again using the improved trajectory. When the desired accuracy in the improvement of the trajectory has been reached, the design is complete. Figure 2.5, taken from Tang, Mays and Yen, illustrates the process.

An interesting concept in Tang, Mays and Yen is a rational method of evaluating risk in the design of storm sewers. From an evaluation of the uncertainties in the design equations, they have developed equations which express risk level as a function of the safety factor. The safety factor is the mean sewer capacity/mean sewer design flow. The objective function is the sum of the installation costs and the damage costs. The damage costs are a function of the pipe diameter and slope. The designer specifies the acceptable risk level for the problem and the optimization process selects the cheapest solution. Tang, Mays and Yen developed three computer models utilizing this principle and applied them to a 9 sewer serial gravity flow storm sewer system.

Mays and Yen (26) used DP and DDDP for the optimization of branched gravity flow storm sewer systems. The technique that they used for analysing the sewer system consisted in decomposing the system into $2N_B + 1$ serial subsystems where N_B is the number of junctions. The serial subsystems were then analysed in a downstream direction in the normal manner. They applied this technique in both the DP and DDDP programs to a 20 sewer branched network. For comparable accuracy, the DDDP program required 30% of computer time

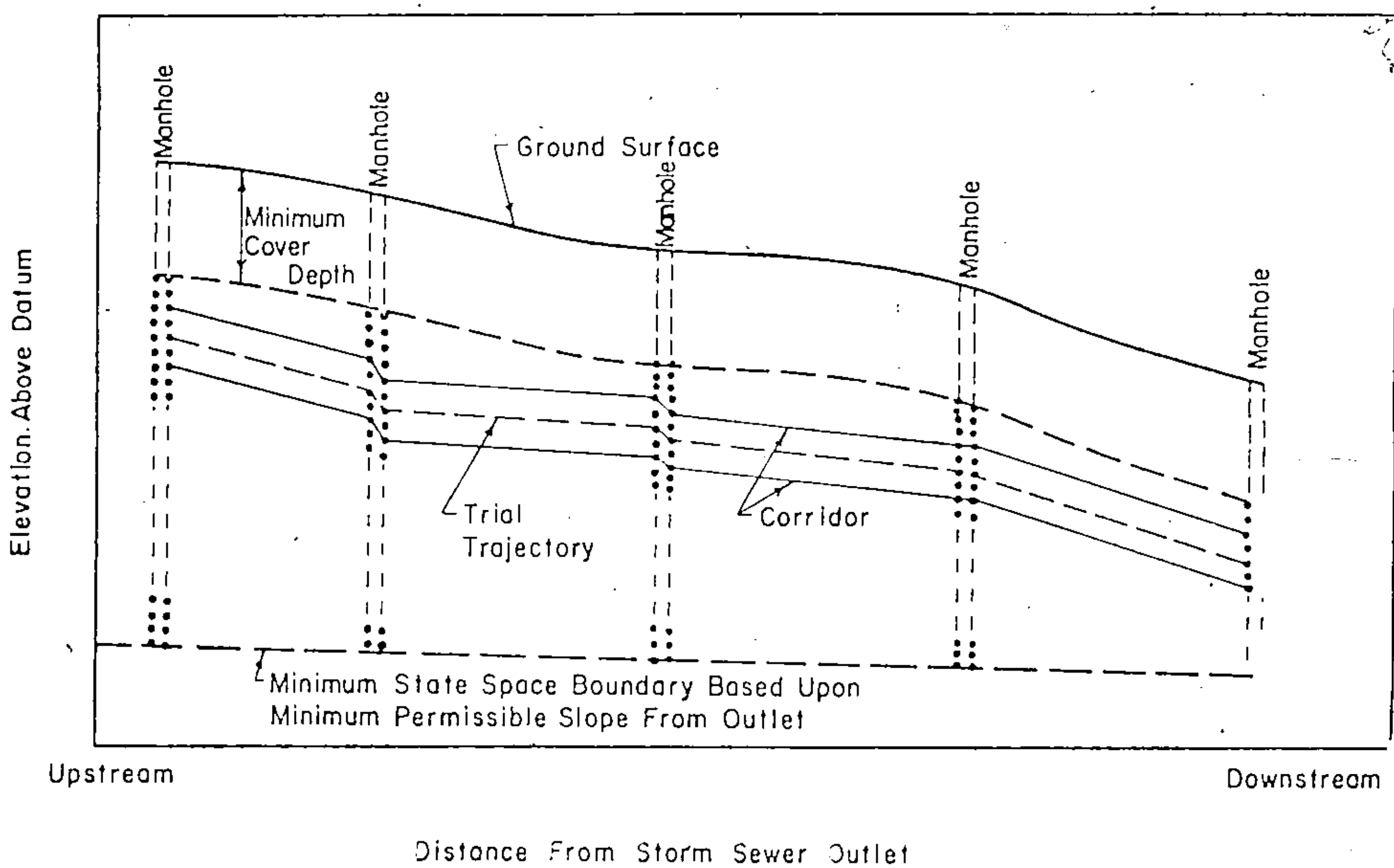


Figure 2.5 STATE SPACE FOR DP AND DDDP PROCEDURES (reproduced from Tang et al (36))

required by the DP program. Global optimization is not assured with the DDDP program if the cost function is a multi-peak function.

Merritt and Bogan (28) stated that their program produced a cost saving of 3% over an as built system and that the program was capable of producing cost savings of 10 - 20%.

Walsh and Brown (39) noted that although the design of sewers is controlled by the fact that excavation and backfill costs represent nearly 80% of the total construction cost the program is useful, particularly in rolling terrain. The program will consider all feasible solutions and their examples showed that cost savings of 6 - 7% could be expected over the conventional design procedure.

3. Other Methods

Geometric programming has been used by Templeman and Wilson (37) but no details are readily available.

Khanna (21) suggests that the "variable metric" method would be a suitable non-linear programming technique. Argaman, Shamir and Spivak (2) replied that they did not agree with this suggestion stating that the large number of layout and vertical profile decision variables made the variable metric computationally difficult.

2.4.4 OPTIMIZATION OF LAYOUT, SEWER SIZES AND VERTICAL ALIGNMENT

Argaman, Shamir and Spivak (1) have developed a dynamic programming solution for the simultaneous optimization of the layout, sewer sizes and vertical alignment for a gravity flow sanitary sewer system whose outlet is predetermined.

In their method a network is prepared showing all the possible sewer connections. An example which is reproduced from Argaman et al's article is shown in fig. 2.6.

The drainage directions of each sewer link are determined from the topography and the designer's judgement. The actual sewer links used in the optimized layout are determined by the optimization process. The selection of the drainage directions fixes the drainage lines which are imaginary lines that join manholes which are the same number of sewers away from the outlet. The method of solution follows the typical dynamic programming formulation described in chapter 3 with the following exceptions:

1. a stage represents all the sewers between adjacent drainage lines, i.e. the optimization process goes from drainage line to drainage line
2. decision variables are the head allocated to each link and the drainage directions.

The program starts at the most upstream stage and proceeds downstream selecting the connectivities and cumulating costs.

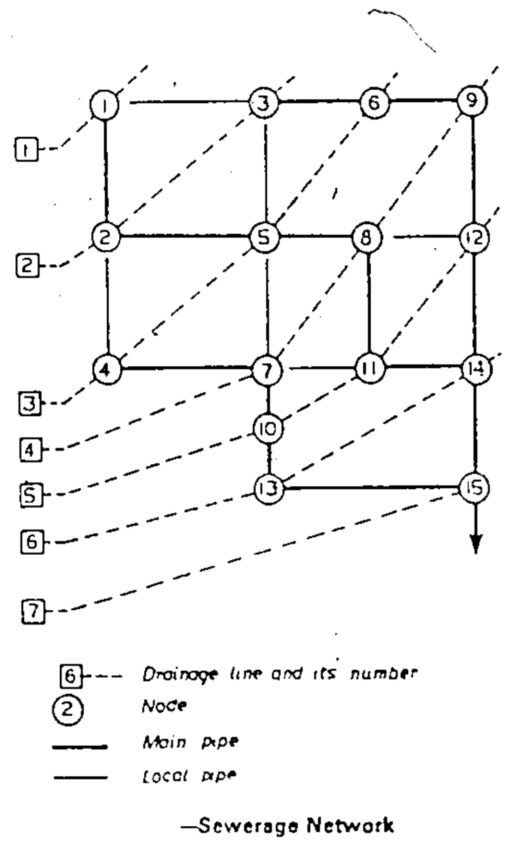


Figure 2.6 ARGAMAN et. al (1) SEWERAGE NETWORK

The program was tested on a hypothetical 10 node, 5 drainage line gravity flow sanitary sewer network and also an actual 36 node, 10 drainage line gravity flow sanitary sewer system.

The authors noted that the program is limited by the large amount of computer time and computer storage required. For example, the first problem requires 25 min. of computer time if four elevations are checked at each manhole. If five elevations are checked, computer time increases by a factor of 10. The program, run on an IBM 360/50, required 50,000 words of core and 500,000 words of disk space.

The authors noted that their formulation did not give an optimum solution because the cumulated costs are brought forward as a function of elevation. No solutions are brought forward which allow consideration of different connectivity. The authors stated that in a typical 20 link network a program to consider different connectivities would require 250 times as much computer space as the program used. This is not feasible with present day computers.

The authors concluded that their program was only suitable for small networks. Large networks could be analysed by decomposing into sub-systems.

2.4.5 CONCLUSION

Although the cost of excavation and backfill amounts to about 80% of the total construction cost of a sewer network and hence minimum dig is the major cost criterion, the application of optimization techniques to fixed layout networks indicates that savings up to 10% may be achieved. Savings may be increased if other network layouts are considered.

It is recognised that these cost savings may be anticipated values. They cannot be considered real values because of the uncertainty attached to the unit prices used for the optimization process.

It appears that the simultaneous optimization of layout, sewer size and vertical alignment is not feasible for large networks due to the large amount of computer space and execution time required.

Methods are available to obtain a good sewer layout. These methods give some consideration to the hydraulics of the network.

The two most common techniques adapted for the optimization of sewer sizes and vertical alignment are linear and dynamic programming. The linear programming researchers have not produced a program which will solve a problem of any size because of the high computer storage and execution times required. DP and DDDP programs appear to be the most practical. DP programs have been developed which can handle a 200 manhole network.

One prime advantage of the optimization techniques is that they permit sensitivity analysis to be performed on the design criteria.

Further research to improve sanitary sewer design could be directed to transient flow conditions and the probabilistic concept of sewage flow.

CHAPTER 3.0 DYNAMIC PROGRAMMING

In this chapter the application and computational efficiency of dynamic programming with respect to sewer design is discussed.

The concept of applying dynamic programming to sewer design has been attributed to Haith (16). Haith considered the head available across the sewer system to be a scarce resource which has to be allocated amongst the sewers so as to minimize the total cost of the sewer system. A definition diagram showing the available head for a simple serial system is shown in figure 3.1.

The available head for the system is defined as the difference in the lowest feasible energy line elevation at the downstream end of the outfall sewer and the energy line elevation at the upstream end of the first or stage 1 sewer. The latter elevation is fixed by the minimum cover requirements. For sewers designed by the full flow assumption, the energy line is taken to be the inside top of the pipe. For partial flow analysis, the energy line is given by:

energy line elevation = invert elevation + depth of flow
+ velocity head

The optimum outfall energy line elevation may be equal to or greater than the maximum outfall elevation providing the minimum cover

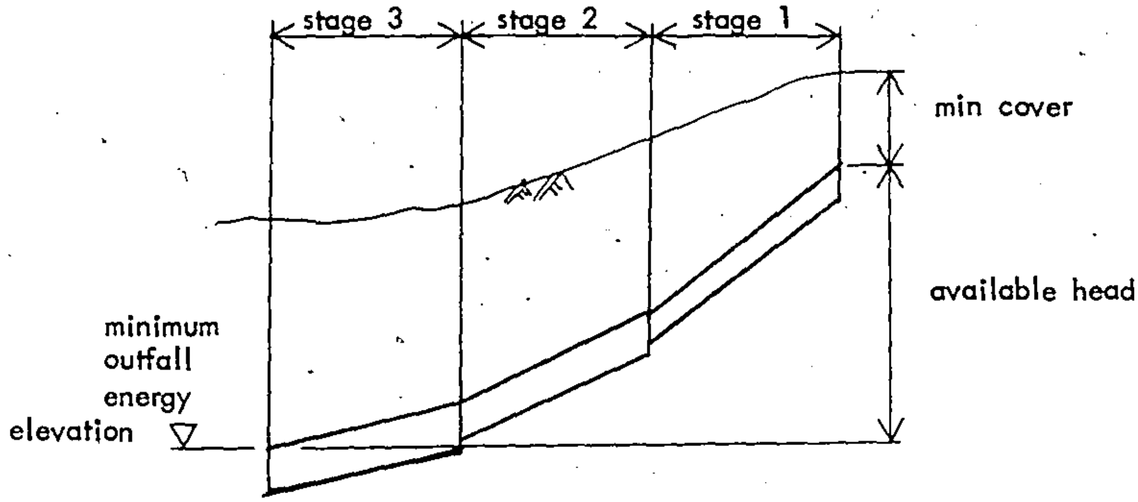


Figure 3.1 PROFILE OF SEWER SYSTEM FOR FULL FLOW CONDITIONS AND SHOWING AVAILABLE HEAD

requirement is adhered to. The available head or part of the available head is allocated to stages 1, 2 and 3 to obtain sewers which meet the design criteria. There will be a solution for each outfall energy line elevation considered. The cheapest solution is selected.

The dynamic programming process starts at stage 1 and proceeds downstream, i.e. it is an initial value problem. At each stage, several values of energy line elevation, or state, at the downstream end of the stage are considered. For each state value, several values of decision, or allocated head, are considered. For each feasible, state and decision combination the smallest suitable sewer is selected; in addition, the accumulated cost of the sewer system to that state is calculated. For each value of state, that solution which gives the cheapest cumulated cost is selected as being the best solution. The decision and the cumulated costs for the best solutions are stored. The procedure is repeated until the final or outfall stage is reached. At the outfall stage, the cheapest system cost is selected and the process is retraced upstream to retrieve the optimum solution for each stage. The method is shown diagrammatically in figure 3.2. In this figure three downstream state values are considered at each stage. The cheapest cumulated sewer system cost to the state is circled. The best sewer location is indicated by the line; the sewer size is above the line, and the stage cost is below the line. The values shown in the figure are for the purposes of illustration only. The optimum solution is indicated by the heavy line.

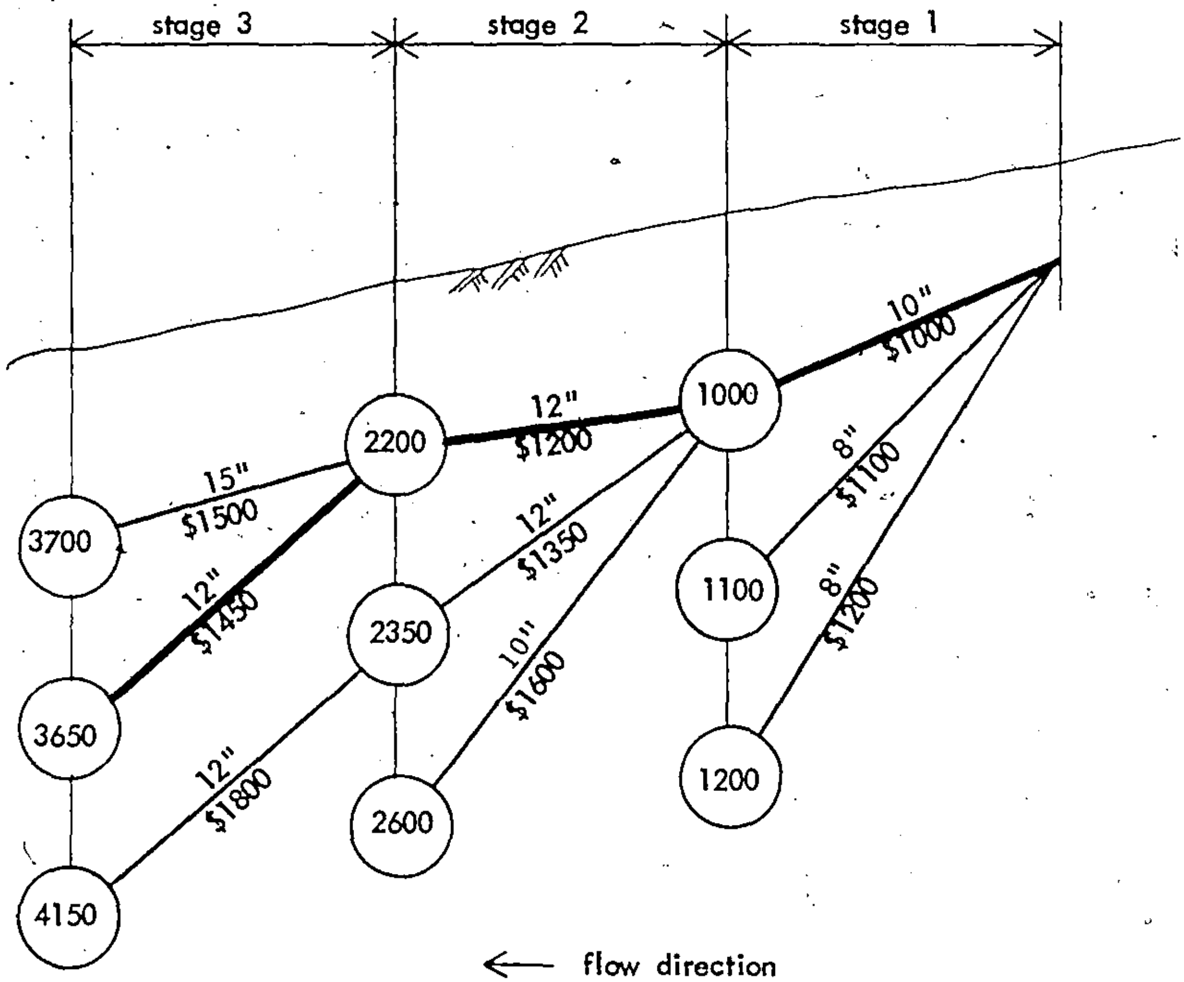


Figure 3.2 DIAGRAM SHOWING OPTIMUM SOLUTION CALCULATION

In the terms of dynamic programming, the design of a sewer system may be stated as follows and is illustrated in fig. 3.3.

Objective:	To minimize the total cost of the sewer system.
Stage:	Sewer + manhole(s).
State:	Sewer energy line elevation (NS).
Input:	Upstream state.
Decision:	Amount of head allocated to the sewer (ND).
Output:	Downstream state = upstream state + decision.
Return:	Cumulated cost of the sewer system. (FSBEST).

The computational advantage of dynamic programming over an exhaustive search technique lies in the principle of optimality. In Bellman's (5) words, this principle states:

> "An optimal policy has the property that whatever the initial state and decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision."

This means that only the best solution to each state need be considered in the optimization of subsequent stages. The exhaustive search technique requires that all the solutions be considered. The computational efficiency of dynamic programming over exhaustive search will be shown by comparing the number of additions and comparisons required by both methods for a simple problem.

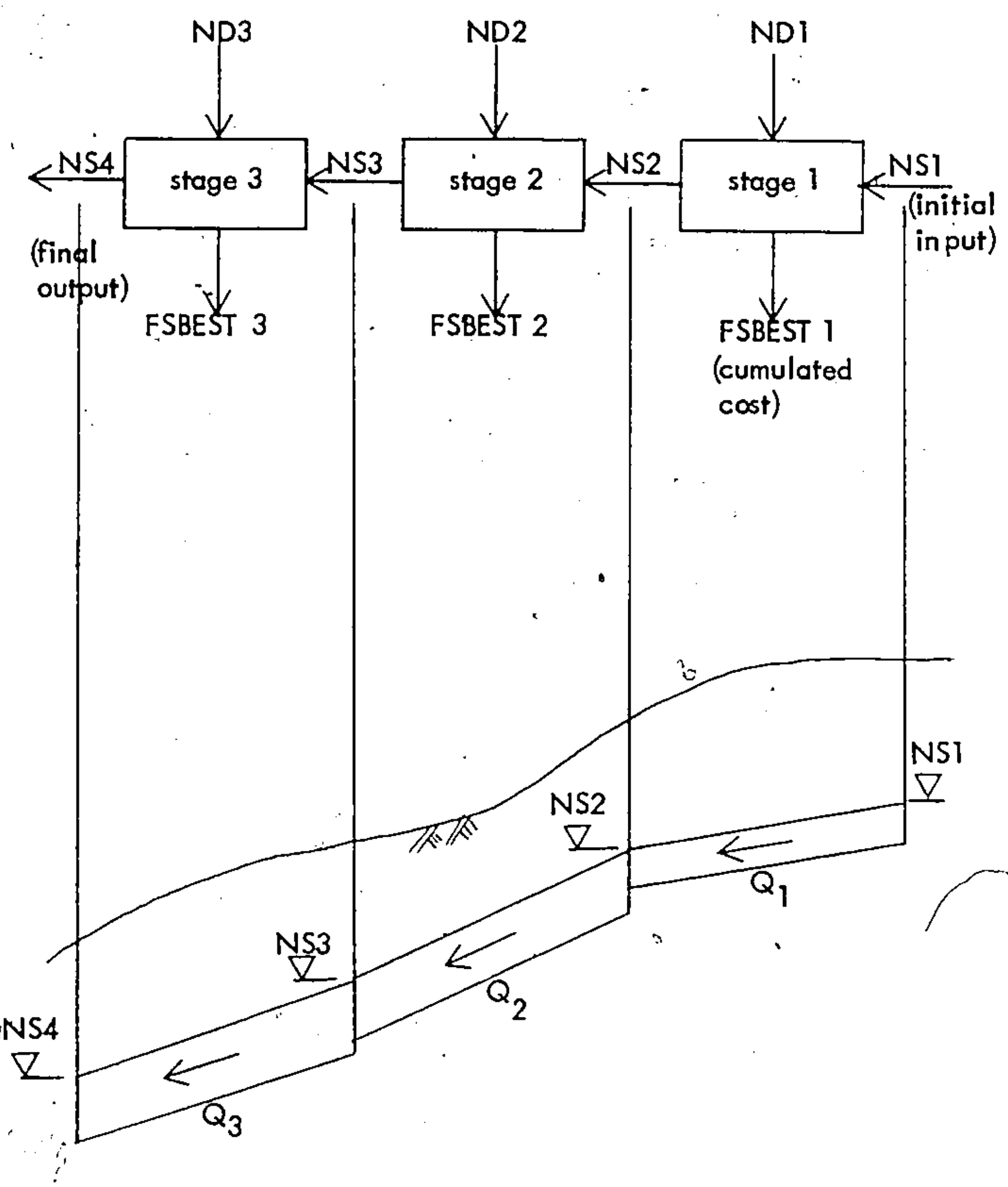


Figure 3.3 THREE STAGE SEWER SYSTEM
(full flow analysis)

Consider a 3 stage problem with 4 possible states for each stage. The problem is shown diagrammatically in figure 3.4. The state and decision increments are equal, e.g. 1 foot.

The feasible decisions for each stage are given in the following table.

Table 3.1 Feasible Decisions

Stage	State at Downstream End of the Stage	No. of Feasible Decisions
1	1	1
	2	1
	3	1
	4	1
2	1	1
	2	2
	3	3
	4	4
3	1	1
	2	2
	3	3
	4	4

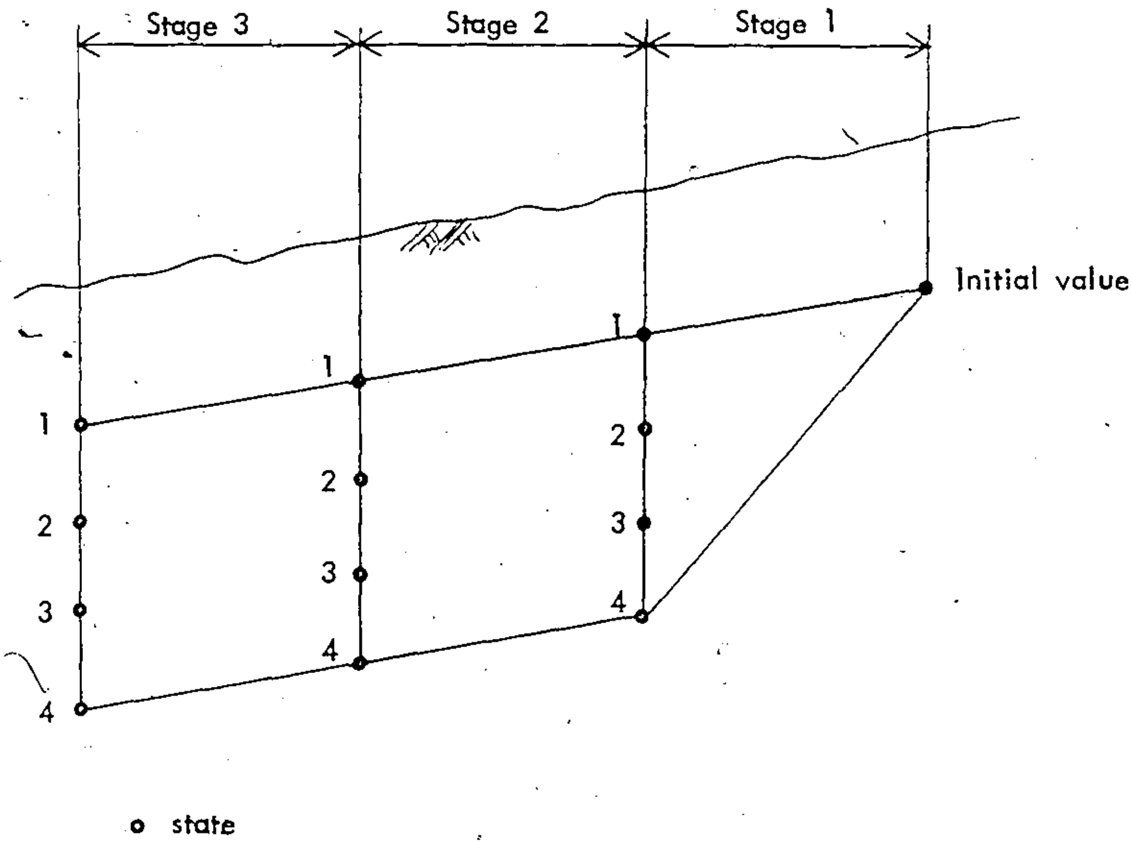


Figure 3.4 DIAGRAM TO ILLUSTRATE COMPUTATIONAL ADVANTAGE OF DYNAMIC PROGRAMMING OVER EXHAUSTIVE SEARCH TECHNIQUE

Exhaustive Search

In an exhaustive search, the number of feasible solutions for each outfall state is given by the following table.

Table 3.2 Feasible Solutions

Outfall State	Feasible Solutions	
1		1
2	1 + 2	= 3
3	1 + 2 + 3	= 6
4	1 + 2 + 3 + 4	= 10

Thus for an exhaustive search technique, a total number of 20 solutions must be examined.

To determine the total cost of each solution, 3 costs must be added. Assuming these costs are added two at a time,

$$\text{number of additions} = 2 \times 20$$
$$= 40$$

To find the cheapest cost, all the final costs must be compared. Hence,

$$\text{number of comparisons} = 19$$

Therefore,

$$\text{number of calculations} = 40 + 19$$
$$= 59$$

Dynamic Programming

$$\text{number of additions} = 2\ (1 + 2 + 3 + 4)$$
$$= 20$$

$$\text{number of comparisons} = 2\ (1 + 2 + 3)$$
$$= 12$$

Therefore,

$$\text{number of calculations} = 32$$

The results show that even in this simple problem a saving in calculations of 46% is made when the dynamic programming technique is used.

In a more general case, let's consider that there are N stages and S feasible states in each stage, and, as in the previous example, the state and decision increments are equal; and state 1 is one increment below the state 1 for the upstream stage.

For the exhaustive search technique, the number of computations is:

$$N \sum_{X=0}^{S-1} (S - X)\ (X + 1) - 1$$

For dynamic programming, the number of computations is:

$$2\ (N - 1) \sum_{X=1}^{S} X - (N - 1)S$$

Suppose we have a problem with 100 stages and 10 feasible states in each stage.

For exhaustive search,

$$\text{number of calculations} = 100 \sum_{X=0}^{9} (10 - X)\ (X + 1)$$

$$= 21,999$$

For dynamic programming,

$$\text{number of calculations} = 2\ (100 - 1) \sum_{X=1}^{10} X - (100 - 1)\ 10$$

$$= 9,900$$

The computational savings are not as spectacular as in other dynamic programming applications (31) because the number of states and decisions is small. It should be noted too, that if state 1 was more than one increment below the upstream state 1, there would be more exhaustive search solutions; however the number of dynamic programming solutions would remain the same.

The choice of head as the 'decision' in the dynamic programming process appears to be the best method of applying dynamic programming to sewer system design. It permits the states and decisions to be incremented uniformly and this permits easy storage and retrieval of data. If, for instance, pipe diameter was the decision variable, non-uniform state and decision increments would occur and this may make it difficult to store and retrieve data.

In conclusion it may be stated that sewer system design may be solved by dynamic programming by treating head as a scarce resource and allocating the available head amongst the sewers to obtain the cheapest cost. The method offers computational advantages over an exhaustive search technique.

CHAPTER 4.0 PROGRAM DESCRIPTION

4.1 OBJECTIVES IN WRITING THE PROGRAM

The objective in writing the program was:

1. to produce a program which would design gravity flow sanitary sewers for a fixed branched sewer layout,
2. to provide dimensional flexibility to suit user requirements,
3. to provide a simple identification system for entering the line data, and storing and retrieving design data,
4. to make it simple for the program user to enter or select design criteria and unit cost data,
5. to provide for drop manholes and for mandatory sewer elevations.

4.2 INTRODUCTION

The program permits the user to enter his own design and cost criteria either as data or as functions.

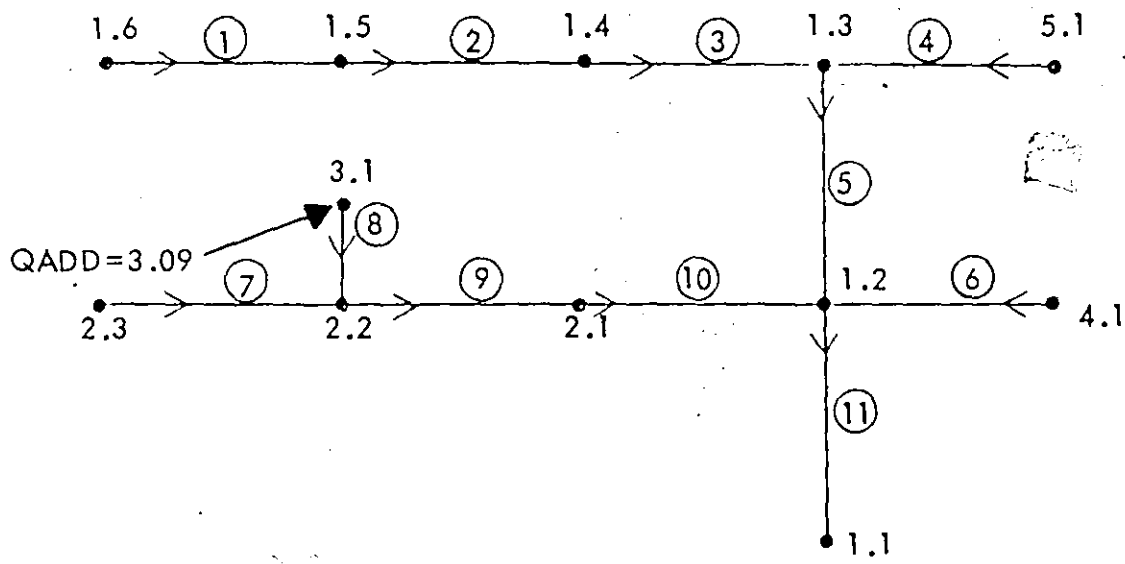
An important feature of the program is the manhole numbering system, the branch numbers and the order of entering the line data. This will be described with reference to figure 4.1.

A decimal system of manhole numbering was adopted. The integer part of the upstream manhole number is the branch number. The sewer number or stage number is implied by the order of entering the line data.

Manhole numbers are assigned thus. The outfall manhole is numbered 1.1. Manholes on the main sewer line, i.e. branch no. 1, are numbered 1.2, 1.3, etc., as one proceeds upstream. It is immaterial which branch is selected as the main sewer line. When all the manholes on branch no. 1 have been assigned numbers, branch no. 2, the next upstream branch is designed. It is immaterial whether the branch to the left or right of manhole no 1.2 is assigned as branch no. 2. The process is repeated working in an upstream direction until all the manholes in the network have been numbered.

This numbering system has the following features:

1. manholes are easily located,
2. integer part of the upstream manhole number is the branch number,
3. comparison of upstream and downstream manhole number for a line enables junction manholes to be identified.



1.1 manhole number

(1) line number

Figure 4.1 NETWORK FOR TEST PROBLEM 1

The order of entering the line data, i.e. manhole numbers, ground elevations, etc., is as shown by the line numbers in figure 4.1. The line data is stored in arrays; the position in the array is given by the line number. To ensure proper storage of design data on the scratch file, for sewers upstream of a junction the sewer on the lower numbered branch is assigned the lower numbered line number, e.g. line nos. 8 and 7 are not interchangeable.

Solution data is stored in the program as a function of the state. States are measured positive downwards from a state datum which is the highest feasible energy line elevation in the system. The best decisions for each state and stage are stored on the scratch file in a location identified by the state and stage.

The data entered by the user includes the design criteria, cost data and collection system data. The collection system data may include any mandatory sewer elevations. The program locates the state datum and reduces all the elevations to state values which are measured positive downwards from the state datum. Design commences at the highest numbered branch and finishes on branch no. 1. Costs are cumulated as the design proceeds downstream. At the outfall stage, the cheapest solution and corresponding state is selected. The optimum decisions are then retrieved from storage and the optimum design is finalized. A two pass method is used. In the first and second passes, the state and decisions are incremented in 1 ft. and 0.1 ft. units respectively. The first pass defines the area of search in the second pass.

Generally, values which are common to more than one function or subroutine are passed through a common file to save storage.

The program consists of a program, two subroutine subprograms and eight function subprograms as listed below:

1. program SEWER
2. function PRICEP (CUT, DIA, PLNGTH)
3. function PRICED (HEIGHT, DIA)
4. function PRICES (HEIGHT)
5. subroutine DESIGN (COSTJ, DGELEV, DTEMP, ELCU, ELCL ELELEV, FSBEST, FSOPT, ISTAGE, IELCU, IELCL, IDBEST, JNS, KTIES, KDEC, KSTAGE, PDIA, PENERGY, PCOST, PSLOPE, PQFULL, PVFULL, PVMIN, PVMAX, PDEPTH, PDROP, PMHLSS, PINVDP, PFALL, PUINV, PDINV, PIPEJ, SDIAM, SDIAMJ, UGELEV, XJN, LINES,NJMAX, NBRNCH, NSMAX, LAYMAX, NRUN)
6. function LOCATE (IJ, LOWEST, INCRMT)
7. subroutine FLOW
8. function PIPE1 (KN, KX, SDIA)
9. function PIPE2 (KN, KX, SDIA)
10. function CSTPM (KN, KS, KX)
11. function COSTM (KKN, UUCUT, DDCUT)

Program SEWER and functions PRICEP, PRICED and PRICES are supplied by the user.

Given any configuration of the sewage network, the program will design the sewer system to provide the cheapest cost.

The program was developed from a serial general dynamic programming code contained in Kuester and Mize (22). The program was written for use on a CDC 6400 computer. One scratch file is required.

It should be noted that the flowcharts included in the appendix are intended to show the ideas and logic of the program. For complete details, the reader should refer to the program listing.

4.3 PROGRAM SEWER

The function of this program is to:

1. set up the necessary storage arrays for use in subroutine DESIGN and assign values to integers used in the dimension stagement.
2. read in data defining user identification, the number of problems, and cost data. A flag integer ICOST is used to specify whether the costs are defined in terms of data points or as function routines.
3. call the main program, subroutine DESIGN for each problem to be solved.

4.4 USER SUPPLIED COST FUNCTIONS

The following functions are used to compute costs if ICOST = 1.

Function PRICEP is used to compute the supply and installation cost of the sewer.

Function PRICED is used to compute the supply and installation cost of the drop pipe.

Function PRICES is used to compute the cost of the manhole shaft.

Details of these functions will be found in the program listing in the appendix.

4.5 SUBROUTINE DESIGN

Subroutine DESIGN is the main program. It is called for each problem.

The function of subroutine DESIGN can be divided into three sections:

1. input section,
2. design section,
3. recovery and output section.

1. Input Section

In this section all the relevant data for the problem are entered. This includes the problem name, hydraulic and physical criteria and sewer system data.

One of the hydraulic input criteria is the value of the variable, SIGN3. If SIGN3 = 0., hydraulic computations are carried out by full flow analysis. In the latter, the sewer is assumed to flow full, the energy line is assumed to be coincident with the inside crown of the sewer and the inside crowns of sewers are made continuous at manholes. If SIGN3 = 3., hydraulic computations are carried out by partial flow analysis and calculations are based on the theoretical energy line. Both methods assume that steady state conditions apply.

2. Design Section

The purpose of the design section is to design the cheapest sewer system subject to the physical and hydraulic constraints imposed on the system.

The subroutine selects downstream states for each sewer and finds the cheapest accumulated sewer costs to suit these states. At the outfall stage, the cheapest or optimum cost is selected, the optimum path is traced upstream and the design is finalized.

The subroutine operates on a two pass system in order to reduce computational time. In the first pass the subroutine uses an incremental value of 1 ft. (100 'units') for state and decision variables. The first pass sets the area of search in the second pass Y feet above and below the first pass optimum energy line, providing there are no limiting constraint elevations. In the second pass, the program defines the state and decision variables in increments of 0.1 ft. (10 units).

The functions of the design section are carried out in the following order.

(i) Calls Subroutine FLOW

The design process commences by calling subroutine FLOW to compute the maximum and minimum flow rate in each sewer. The maximum and minimum flow rates are stored in the arrays QDMAX(I) and QDMIN(I) where I is the sewer number.

Subroutine FLOW is described in section 4.6.

(ii) Assigns Junction Manhole Numbers

Junction manholes are identified by the manhole numbering system. If the integer part of the upstream manhole number is not equal to the integer part of the downstream manhole number, then the latter is a junction manhole. The subroutine searches through the manhole numbers in the sewer system, identifies the junction manholes and stores the junction manhole numbers in the array XJN(NJUNC), where NJUNC is the identification number for the junction.

(iii) Computes the number of sewers in each branch and number of branches

The subroutine determines the number of sewers in each branch by searching through all the upstream manhole numbers. Those with the same integer value are on the same branch and are summed. The number of sewers in each branch is stored in the array KBRNCH(I) where I is the branch number or the integer part of the upstream manhole number.

To ensure proper storage of the best decision and retrieval of the best decisions and retrieval of the optimum decision, the sewer numbers in any branch must be consecutive values. Hence the sewers tributary to any branch must be considered part of that branch but with the exception that they are treated as dummy sewers. This is illustrated for the test problem 1 layout shown in figure 4.1.

For this network

KBRNCH	(1)	=	11	(includes 6 dummy sewers)
"	2	=	4	(includes 1 dummy sewer)
"	3	=	1	
"	4	=	1	
"	5	=	1	

(iv) Selects the highest, furthest upstream ground elevation and the corresponding branch number

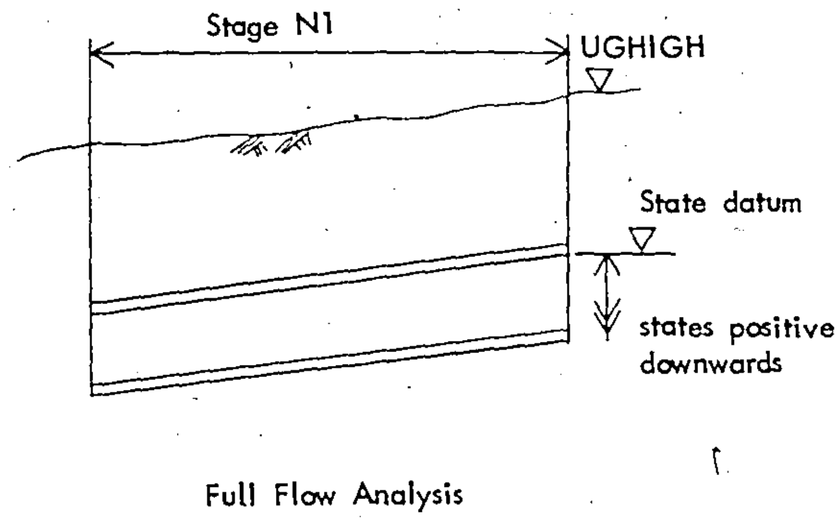
The program searches through all the upstream ground elevations of the first sewers in each branch. The variable UGHIGH is assigned to the highest value. The corresponding branch number is assigned to LLHIGH.

(v) Selects the State Datum

The state datum is the upstream energy line elevation at the highest, furthest upstream manhole.

For full flow design, the energy line is considered to be the inside top of the pipe. For partial flow design, it is not possible to predict the location of the energy line, thus it is assumed that the energy line elevation is at the pipe centre-line.

The state datums are shown diagrammatically in fig. 4.2.



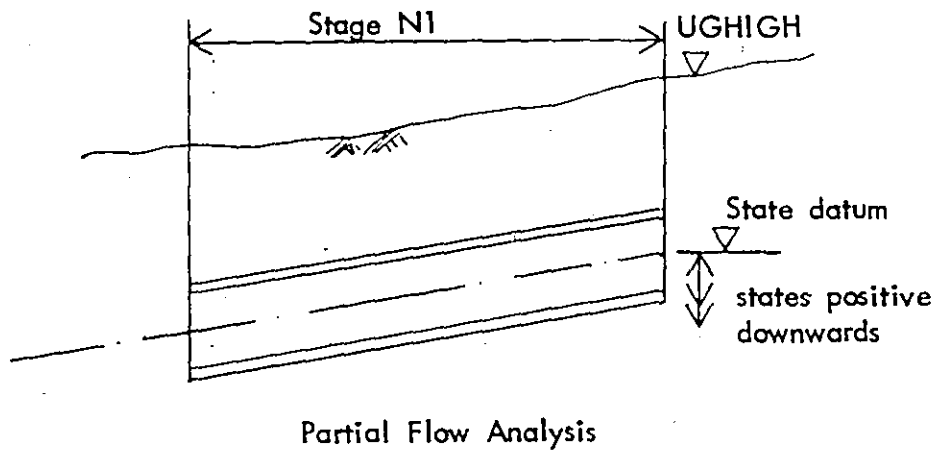


Figure 4.2 DIAGRAM SHOWING LOCATION OF STATE DATUMS

(vi) Converts all the elevations to state values

Ground elevations, constraint elevations and the outfall elevation are reduced to state datum. States are measured positive downwards from the datum and are converted to integer values by multiplying by 100.

If SIGN 4 = 0., all states must be to the nearest hundred unit. If SIGN 4 = 4., all states must be to the nearest ten unit.

(vii) Controls the order of computation of the branches

The order of computation is given by the manhole numbering system. The highest numbered branches are designed first.

The program designs the sewer system by starting on the highest numbered branch, KSER, and finishing on branch #1.

(viii) Determines the lowest, N1 and highest, NSTAGE, sewer numbers for the branch

The program selects the upstream sewer number, N1, and the downstream sewer number NSTAGE, by searching through the upstream manhole numbers and utilising the array KBRNCH(I).

(ix) Controls order of solution in the branch

The program controls the order of solution from the upstream sewer to the downstream sewer in each branch.

(x) Assigns JU, JL and IDBEST for the dummy sewers

Dummy sewers are removed from the actual design process. They are selected by the fact that the value of IUMHN is not equal to the branch number.

To ensure continuity of the design process, the critical upstream state limits are transferred to the next downstream stage.

All the best decisions are set equal to 0.

(xi) Sets the limits for the state variable NS

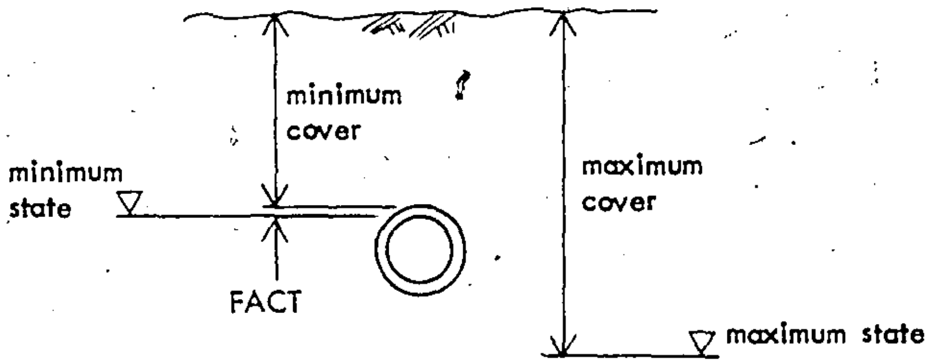
The subroutine determines the minimum and maximum state values at the downstream end of each sewer. The method of computation of these values in the first pass is shown in figure 4.3.

FACT is a variable which is used to try to adhere to the minimum cover requirement. In the most upstream state of any branch, FACT is computed as per the method of locating the state datum. For other stages, FACT is computed from the first feasible pipe computed in the upstream stage.

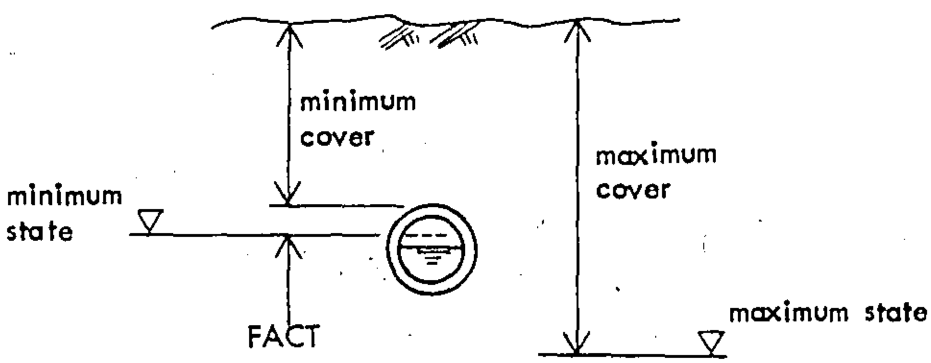
The pipe thickness is assumed to be that of reinforced concrete ASTM c76 B wall.

In the second pass, the limits are obtained from the constraints set Y feet above and below the optimum energy line derived in the first pass.

Any mandatory elevations have priority over the foregoing assignments.



Full Flow Analysis



Partial Flow Analysis

Figure 4.3 DIAGRAM SHOWING METHOD OF COMPUTATION OF STATE LIMITS

(xii) Sets the upstream state limits JU AND JL

The subroutine sets the upstream state limits to ensure that only state-decision combinations which have feasible upstream contributors are used.

For stage N1 sewers, JU and JL are assigned the values of the minimum and maximum upstream states respectively. For other sewers, JU and JL are assigned the values of NS and HS, respectively, for the upstream sewer.

(xiii) Sets the limits for the decision variable ND

The decision variable is the head allocated to each stage in increments of 100 (1 foot) and 10 (0.1 foot) in the first and second pass respectively.

The low decision value in each pass is one increment.

In the first pass, the maximum decision value is the difference between the maximum state at the downstream end of the sewer and the minimum state at the downstream end of the upstream sewer. The latter states are computed as described previously.

In the second pass, the maximum decision value is computed from the state elevation constraints computed in the first pass. The maximum decision value is the difference between the maximum elevation constraint for the sewer and the minimum elevation constraint for the upstream sewer. The subroutine checks to ensure that a branch or dummy sewer is not used in the calculation. The method of computing the maximum decision is shown in fig. 4.4.

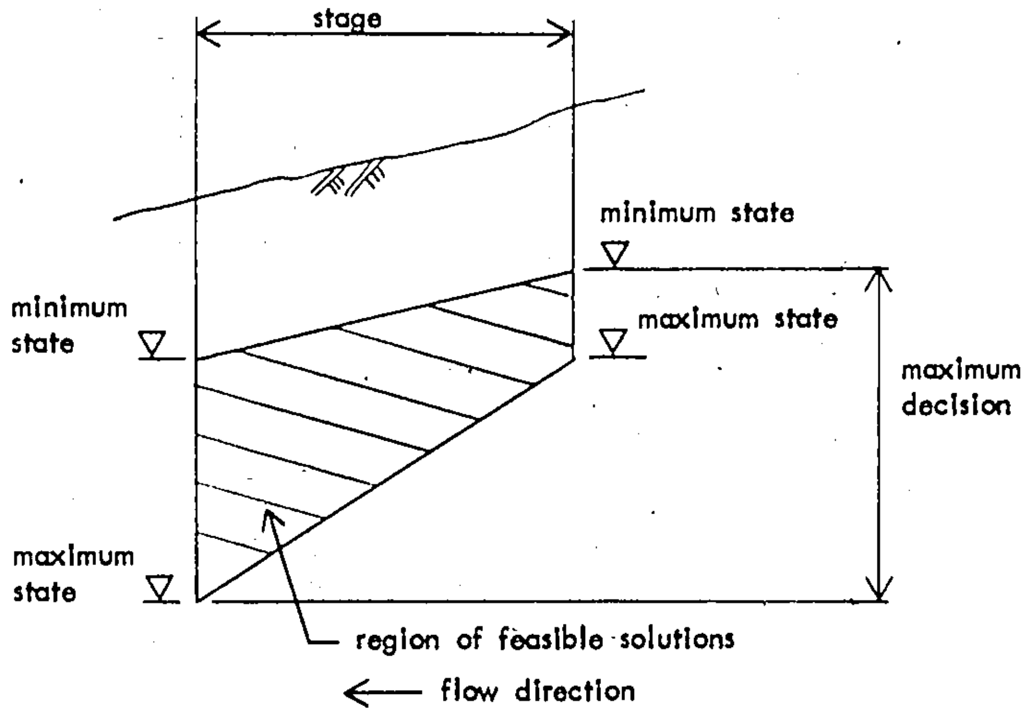


Figure 4.4 DIAGRAM SHOWING METHOD OF COMPUTING MAXIMUM DECISION IN SECOND PASS

(xiv) Sorts Junction Costs

Prior to designing a sewer downstream of a junction the program sorts the branch costs to ensure that the cheapest feasible branch cost and the corresponding diameter is considered for the sewer downstream of the junction.

(xv) Selects feasible state and decision combinations

The subroutine ensures that only those state and decision combinations which have a feasible upstream contributor are considered. This is done by comparing the upstream state, given by the state and decision, with the upstream state limits JU and JL.

(xvi) Computes the best pipe size for each state in the stage

For each feasible state and decision combination, the subroutine: - computes the sewer size by calling either function PIPE1 or function PIPE2; computes the stage cost by calling function CSTPM; computes the cumulated cost FNSXN.

The subroutine then selects and stores the cheapest cumulated cost for that state, the associated decision and pipe size in the arrays FSBEST, IDBEST and SDIAM. Tied solutions are indicated and broken in favour of the lower valued decision. Solutions which have a pipe size greater than the maximum allowable diameter, are discarded.

When all the best solutions for each state are selected, the pipe sizes and cumulated costs are stored in the arrays DTEMP and FSOPT for use in the design of the downstream stage and the best decisions IDBEST are stored on scratch file .

If the stage is the outfall stage for a branch, the pipe sizes and cumulated costs are stored in the arrays PIPEJ and COSTJ respectively.

An example showing the method of storing the best decision for test problem number 1 is shown in figure 4.5.

(xvii) Selects Cheapest Sewer System Cost

The program searches through the outfall costs to select the cheapest, the associated state and decision.

3. Recovery and Output

The function of this section is to retrieve the optimum decisions from the scratch file , re-calculate the optimum pipe data, calculate vertical alignment data and print the results.

The process takes place branch by branch and travels in the opposite direction, with respect to branches, than that of the design process i.e. recovery starts on branch #1 and finishes on branch #KSER.

The process starts by putting the scratch file in reading mode and at the beginning of the location of the stage (NSTAGE -1) best decisions in preparation to read these decisions.

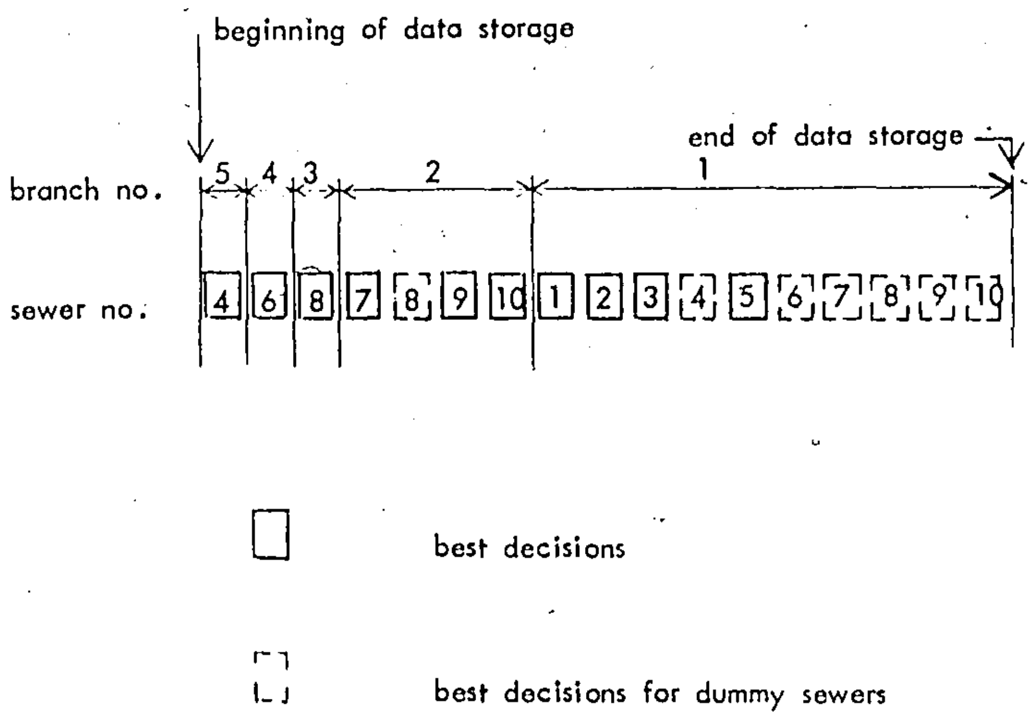


Figure 4.5 DIAGRAM SHOWING ORDER OF STORAGE OF BEST DECISIONS ON SCRATCH FILE

As the optimum outfall state and decision are known, the optimum state for the stage (NSTAGE-1) is known and hence the optimum decision can be located and read from the scratch file. The file is then backspaced to the front of the (NSTAGE-2) best decision location and the process is repeated. This process continues until all the optimum decisions for the branch have been retrieved.

If the downstream manhole of the stage under design is a junction manhole, the state at this manhole is stored in the array JNS(NJUNC) for use in the retrieval of the optimum decisions for the branch sewers tributary to the junction. The state value stored is the maximum possible state value for the branch outlet. A routine ensures that the state value which provides the minimum feasible cost solution for the branch is selected.

When all the optimum decisions for a branch have been retrieved, program execution proceeds to the re-calculation of the optimum pipe data and the calculation of the vertical alignment data. The latter calculations are performed in the downstream direction.

When a dummy sewer, represented by a zero decision, is encountered the subroutine assigns the data for the upstream sewer to the dummy sewer to ensure continuity in design.

The sewer data is stored in arrays during design. On completion of the design of a branch, the recovery process goes to the next upstream branch. When all the branches have been designed the sewer design data is printed.

4.6 SUBROUTINE FLOW

Subroutine FLOW is called from subroutine DESIGN. It computes the design flow rate QDMAX and the minimum flow rate QDMIN for each sewer in the network. QDMAX is the sum of the peak domestic flow rate, and the average commercial, industrial and infiltration flow rates, and QADD. QADD is an additional flow. QDMIN is the sum of the average domestic, commercial industrial and infiltration flow rates, and QADD. The peak domestic flow is obtained by multiplying the average domestic flow rate by a peaking factor.

The subroutine uses the following data from the COMMON file:

ACRES, DMHN, PEAKFS, POPD, QADD, RCOM, RDOM, RIND, RINF, TYPE, UMHN.

A description of these variables will be found in Appendix 2. If PEAKFS = Ø, the following peaking factor formula is used:

$$PKF = (18 + \sqrt{P}) / (4 + \sqrt{P})$$

where,

P = cumulative population tributary to the sewer in thousands.

If PEAKFS = 1, the following peaking factor formula is used:

$$PKF = 5/P^{0.2}$$

The peaking factor is restrained between upper and lower limits PKFMAX and PKFMIN.

4.7 FUNCTION PIPE1 (KN, KS, SDIA)

Function PIPE1 is called from subroutine DESIGN for each feasible state and decision combination to compute the pipe size, if full flow analysis is specified. Function PIPE1 selects the smallest pipe to meet the requirement that the sewer capacity exceeds the maximum design flow QDMAX. The subroutine checks to ensure that the velocity for the maximum flow is not greater than VMAX or less than VMIN. The minimum velocity in the sewer is computed using the minimum flow rate QDMIN; however this velocity is not used as a criterion for design.

The input arguments are the stage number KN, the downstream state KS and the smallest permissible diameter SDIA. The subroutine uses the following values from the COMMON file - DIA, DIAMAX, G, NPIPES, PI, QDMAX, QDMIN, RMANN, RUN, VMAX AND VMIN, a description of these variables will be found in Appendix 2.

Starting with the smallest permissible diameter the function computes the sewer capacity by Manning's equation. The pipe diameter is incremented until the sewer capacity exceeds the maximum design flow. The full flow velocity VFULL, is then computed by Kutter's equation. Kutter's equation is recommended by the Ontario Ministry of the Environment for minimum velocity calculations.

The full flow velocity is compared with the minimum full flow velocity for self-cleansing, VMIN. The pipe size is increased until the minimum velocity requirement is equalled or exceeded.

The full flow velocity, VFULL is then compared with the allowable maximum velocity. If VFULL does not exceed the allowable maximum velocity, the design is completed with the assignments of PIPE1, VMINP, VMAXP, DEPTH and ENERGY. VMAXP is assigned equal to VFULL. VMINP is computed from Pomeroy's equation. DEPTH and ENERGY are both assigned values of zero; actual values are not required because the crowns of sewers are made continuous for full flow conditions.

If VFULL exceeds the maximum allowable velocity, execution proceeds to the drop manhole section of the function. In this section the maximum feasible slope for the pipe size being considered is computed from Manning's equation. The sewer capacity for this slope is computed from Manning's equation; if the sewer capacity is less than the design flow QDMAX, execution returns to the beginning of the function. If the sewer capacity exceeds the design flow, the full flow velocity is calculated from Kutter's equation. If the full flow velocity is less than or equal to the permissible maximum velocity, a check is made to ensure that the head is less than the fall and then the final assignments are made and execution returns to DESIGN.

If the full flow velocity exceeds the maximum allowable velocity, the sewer slope is decreased subject to the slope exceeding the minimum slope. The drop manhole routine is then executed with the new value of slope.

A definition diagram for a drop manhole is shown in figure 4.6.

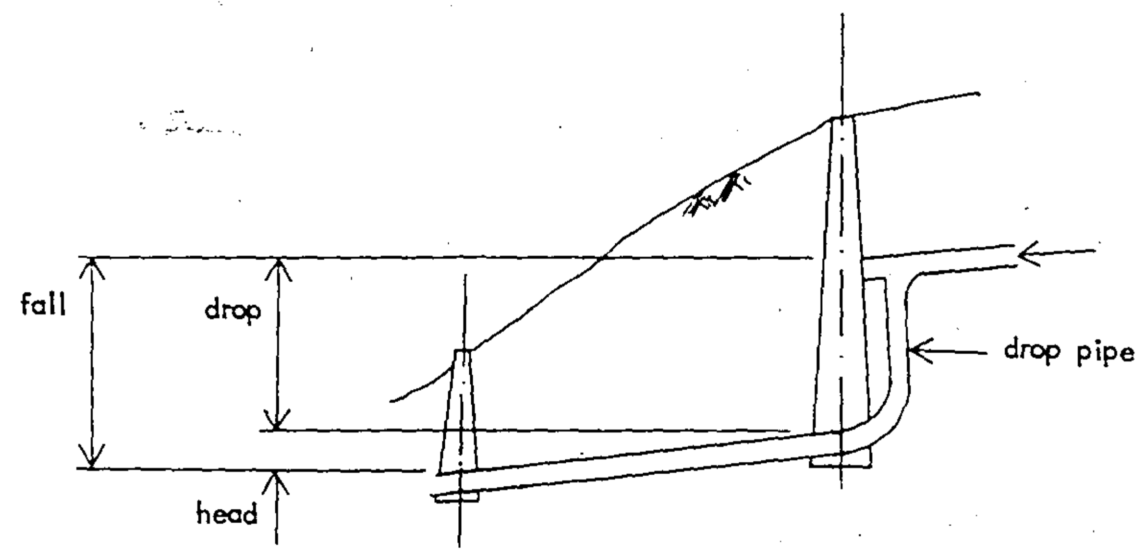


Figure 4.6 DEFINITION SKETCH FOR DROP MANHOLE
(full flow analysis)

In the event that the design requirements cannot be met, function PIPE1 sets PIPE1 equal to the default pipe size, DIA(NPIPES).

Subroutine DESIGN discards any solutions incorporating the default pipe size.

4.8 FUNCTION PIPE2 (KN, KS, SDIA)

Function PIPE2 is called from subroutine DESIGN for each feasible state and decision combination to compute the pipe size if partial flow analysis is specified. The function selects the smallest pipe to meet the requirement that the sewer capacity exceeds the maximum design flow QDMAX and to meet the velocity requirements.

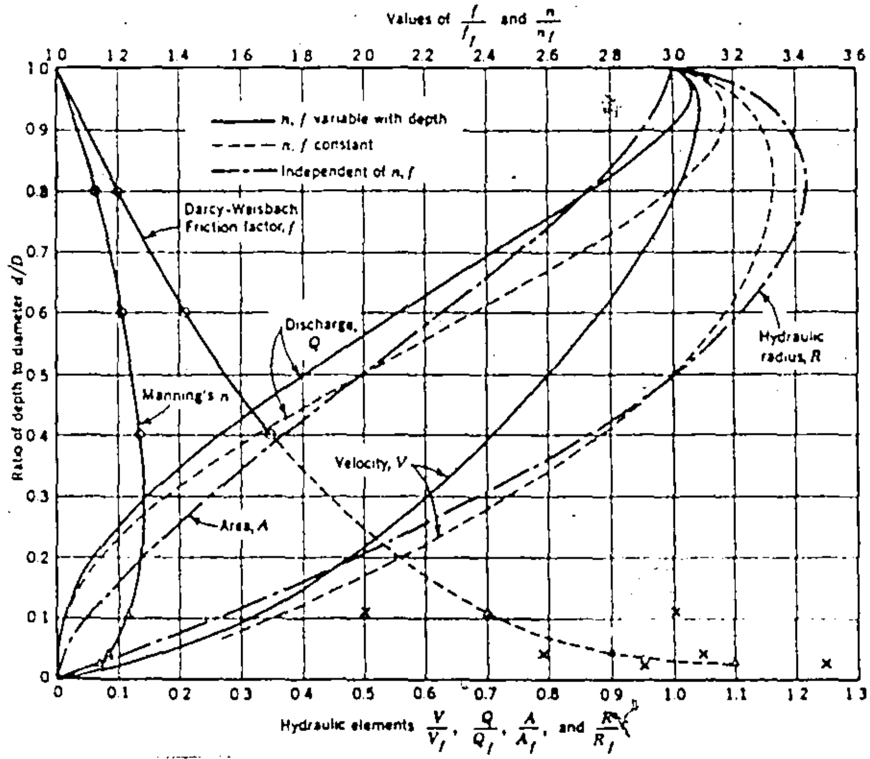
Function PIPE2 is similar to function PIPE1 with the following exceptions:

1. partial flow conditions are analysed,
2. the maximum velocity constraint VMAX is assumed to pertain to the actual velocity in the sewer for the QDMAX flow rate, rather than the full flow velocity,
3. sewers are designed to ensure a velocity equivalent in self-cleansing action to that of the same size sewer flowing full at velocity VMIN,
4. Manning's equation is used throughout,
5. Manning's roughness coefficient is assumed to vary with depth.

For computational efficiency a semi-graphical method was adopted for computing hydraulic properties.

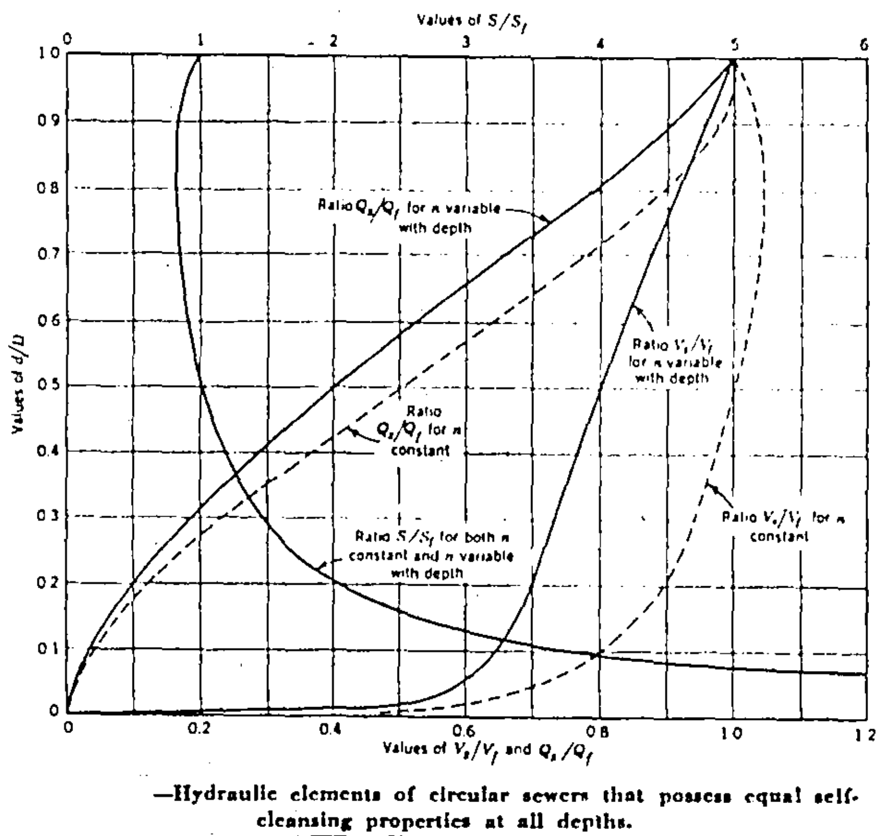
The actual velocity in the sewer is computed by means of the hydraulic elements chart shown in figure 4.7a. This chart is reproduced from the Design and Construction of Sanitary and Storm Sewers (12). The d/D

(a)



—Hydraulic-elements graph for circular sewers.

(b)



—Hydraulic elements of circular sewers that possess equal self-cleansing properties at all depths.

Figure 4.7 HYDRAULIC - ELEMENTS GRAPHS

ratio is obtained from the Q/Qf ratio. The v/vf ratio is obtained from the d/D ratio. Hence v is obtained.

The sewer slope to obtain equal self-cleansing action is computed by means of the hydraulic elements chart shown in figure 4.7b. The ratio S/Sf is obtained from the ratio Q/Qf and hence S is obtained. The actual slope is then compared with the value S.

4.9 FUNCTION CSTPM (KN, KS, KX)

Function CSTPM is called from subroutine DESIGN for each feasible sewer size considered for a stage. The function returns the cost of the sewer and manholes for the stage.

The input arguments are the stage number KN, the downstream state KS and the head allocated to the sewer KX. The function uses the following variables from the COMMON file - DROP, ENERGY, ICOST, IDGS, IUGS, PIPE, RUN, SIGN3. A description of these variables will be found in Appendix 2.

The function computes the average cut or depth to subgrade. Allowance is made for drops. The cost of the pipe is then computed from the cost array CPLF or the cost function PRICEP depending on the value of the signal ICOST.

CSTPM calls function COSTM to compute the appropriate manhole costs.

4.10 FUNCTION COSTM (KKN, UUCUT, DDCUT)

Function COSTM is called from subroutine CSTPM. The function returns the cost of the manholes for the sewer.

The input arguments are the stage number KKN, the upstream cut UUCUT and the downstream cut DDCUT. The function uses the following variables from the COMMON file - CBOX, DCLF, DROP, G, ICOST, NSTAGE, N1, PIPE, SCLF. A description of these variables will be found in Appendix 2.

For the uppermost sewer in any branch, the cost of the upstream and downstream manhole is calculated. For the most downstream sewer in any branch, there are no manhole costs. For other sewers there is a downstream manhole cost only.

The manhole cost is the sum of the manhole chamber CBOX plus the shaft cost.

Where a drop manhole is required, other than where a branch outlet joins a junction manhole, the cost of the downstream manhole for the previous stage must be updated. The extra cost for the shaft and drop pipe is included in the cost for the stage under design. If the drop is less than 2'-0", the drop pipe cost is set equal to zero.

No drop pipe costs are included for a branch outlet pipe which enters a junction manhole at an elevation higher than the main sewer.

Shaft costs and drop pipe costs are computed from cost arrays or cost functions depending on the value of the signal, ICOST.

4.11 FUNCTION LOCATE (IJ, LOWEST, INCRMT)

Function LOCATE returns the identification number for the state. The number enables states to be identified and permits the storage and retrieval of the best decisions.

4.12 MATHEMATICAL EQUATIONS

This section describes the mathematical methods used to compute the following:

1. velocity in a sewer,
2. sewer slope to prevent deposition of solids,
3. energy losses at manholes,
4. vertical alignment at manholes.

1. Velocity in a Sewer

The program computes the average velocity of flow in a sewer by the following methods:

(i) Kutter's formula,

(ii) Manning's formula,

(iii) Hydraulic-elements,

(iv) Pomeroy's formula.

(i) Kutter's Formula

The Ontario Ministry of the Environment recommends the use of Kutter's formula when computing the minimum velocity in full flow analysis.

The formula states:

$$v = c\sqrt{RS}$$

In which, v = velocity of flow (f.p.s.)

c = Kutter's friction coefficient

R = hydraulic radius (fr)

S = slope of energy line

The value of c is given by :

$$c = \frac{41.65 + \frac{0.00281}{S} + \frac{1.811}{n}}{1 + \frac{n}{\sqrt{R}}\left(41.65 + \frac{0.00281}{S}\right)}$$

in which, n = Manning's roughness coefficient.

(ii) Manning's Formula

Manning's formula is similar to Kutter's formula except for:

$$c = \frac{1.486}{n} R^{1/6}$$

Thus the Manning formula is:

$$v = \frac{1.486}{n} R^{2/3} S^{1/2}$$

(iii) Computation of Flow Conditions in Partly Filled Circular Sewers

Flow conditions in partly filled sewers may be obtained from the relationships obtained from Fair, Geyer and Okun (14),

$$v/vf = (nf/n)(R/Rf)^{2/3}$$

$$Q/Qf = (nf/n)\ (A/Af)(R/Rf)^{2/3}$$

in which,

v = velocity

n = Manning's roughness coefficient

R = hydraulic radius

Q = flow rate

R = hydraulic radius

The above terms refer to the partly filled section; the terms with the suffix f refer to the corresponding full section.

For convenience, the hydraulic-elements graph for circular sewers in Design and Construction of Sanitary and Storm Sewers (12) was utilised to solve the above equations. This graph is reproduced in figure 4.7a.

Data points, at intervals of d/D = 0.05, were interpolated from the curve expressing the relationship:

$$d/D \text{ vs } Q/Qf$$

in which d and D represent the depths at partial and full flow respectively. These data points were used in a least squares polynomial curve fitting computer program (20) to obtain the following relationships,

$Q/Qf < 0.14$,

$$d/D = 6.23\,(Q/Qf) - 59.23\,(Q/Qf)^2 + 214.44\,(Q/Qf)^3$$

$0.14 \leqslant Q/Qf < 0.9$,

$$d/D = 0.2 + 0.73\,(Q/Qf) - 0.04\,(Q/Qf)^2$$

Maximum sewer capacity was assumed to be achieved at a d/D value of 0.9. The theoretical increase in capacity over the full flow capacity in the d/D range of 0.9 to 1.0 was not utilized.

Data points, at intervals of $d/D = 0.05$, were interpolated from the curve expressing the relationship:

$$v/vf \quad vs \quad d/D$$

These points were used in a least squares polynomial curve fitting computer program (20) to obtain the following relationship:

$$v/vf = .018 + 3.47\,(d/D) - 7.27\,(d/D)^2 + 9.04\,(d/D)^3 - 4.28\,(d/D)^4$$

The d/D ratio was obtained from the Q/Qf ratio. The v/vf ratio was obtained from the d/D ratio. Hence velocity v was obtained.

(iv) Computation of Minimum Velocity

Pomeroy's formula was used to compute the partial flow velocity in the sewer for the minimum design flow rate QDMIN. The formula was developed by Pomeroy (32) from a series of tests carried out on sewer sizes up to 24-inch diameter. For partly filled pipes of circular section the formula is:

$$v = 1.40KS^{0.41}Q^{0.24}$$

in which,

v = velocity (f.p.s.)

K = velocity coefficient

S = slope of sewer

Q = discharge (c.f.s.)

This formula is discussed in section 5.1.

Pomeroy found that diameter has no significant effect on the velocity. His research provided the following average Manning coefficient and K values:

Table 4.1 Roughness Coefficients

Type of Pipe	Asbestos Cement	Vitrified Clay	Concrete
n (measured)	.0122	.0136	.0165
nf	.0112	.0125	.0151
K	18.9	17.8	15.2

The measured n values were measured with an approximate d/D value of 0.25. The nf value is the corresponding n value for full flow.

An expression relating K to nf was obtained from a linear least squares program on a Hewlett-Packard desk calculator 9100B.

$$K = -956 * nf + 29.66$$

2. Sewer Slope to Prevent Deposition of Solids

There are two methods available in sanitary sewer design to determine the minimum sewer slope required to prevent deposition of solids - the minimum permissible velocity and tractive force methods. The minimum permissible velocity is the minimum velocity to prevent deposition of solids. In this program VMIN is this minimum permissible velocity for a sewer flowing full.

For full flow analysis the velocity at sewer capacity must equal or exceed VMIN; for flows less than sewer capacity it is assumed that the velocity is sufficient to prevent deposition of solids.

For partial flow analysis, the slope of the sewer must be such that, at the maximum design flow rate QDMAX, the tractive force must be at least equal in self-cleansing to that of a sewer flowing full at velocity VMIN. This assumes that equality of tractive force means equality of cleansing. The tractive force equation is:

$$T = \gamma RS$$

where,

T = boundary shear stress (p.s.f.)

γ = specific weight of the sewage (p.c.f.)

R = hydraulic radius of flow area (ft)

S = slope of sewer

From the tractive force equation, the following equation may be derived to compute a slope to ensure equal self-cleansing for partly filled sewers:

$$\frac{Qs}{Qf} = \frac{nf}{n} \frac{A}{Af} \left(\frac{Sf}{S}\right)^{1/6}$$

in which,

Qs = flow rate in partly filled sewer

Qf = sewer capacity

Sf = slope of sewer flowing full at minimum velocity

S = slope of partly filled sewer

A = flow area

Af = flow rate flowing full

nf = full flow Manning's n

n = partial flow Manning's n

For convenience, the hydraulic elements chart for circular sewers that possess equal self-cleansing properties at all depths as shown in Design and Construction of Sanitary and Storm Sewers (12) was used to solve the above equation. This chart is reproduced in figure 4.7b.

Data points, at intervals of d/D from 0.125 to 0.5, were interpolated from the curves:

d/D vs Qs/Qf

d/D vs S/Sf

These points were used in a least squares polynomial curve fitting computer program (20) to obtain the following relationships:

$Qs/Qf < 0.19$

$$S/Sf = 9.97 - 269.29\,(Qs/Qf) + 3540.06\,(Qs/Qf)^2 - 20348.2\,(Qs/Qf)^3 + 41872.5\,(Qs/Qf)^4$$

$Qs/Qf \geq 0.19$

$$S/Sf = 2.23 - 6.26\,(Qs/Qf) + 12.46\,(Qs/Qf)^2 - 13.19\,(Qs/Qf)^3 + 5.74\,(Qs/Qf)^4$$

3. Energy Loss Computations for Partial Flow Analysis

Energy losses at manholes, are computed from the formulae given by Davis and Sorenson (8).

for increasing velocity transitions,

$$\Delta E = 0.2\left(\frac{v_L^2}{2g} - \frac{v_U^2}{2g}\right)$$

for decreasing velocity transitions,

$$\Delta E = 0.3\left(\frac{v_U^2}{2g} - \frac{v_L^2}{2g}\right)$$

in which,

ΔE = head loss (ft)

v_U = velocity before transition (f.p.s.)

v_L = velocity after transition (f.p.s.)

A minimum loss of 0.02 foot was adopted. Davis and Sorenson advise that junction manholes be treated as two or more transitions with computations being made separately for each sewer. This feature was not included in the program.

4. Vertical Alignment at Manholes

In full flow analysis, vertical alignment is achieved by making the crowns of the incoming and outgoing sewers continuous.

In partial flow analysis vertical alignment is achieved by making the energy grade line continuous, taking into account the energy loss ΔE at the manhole. This is achieved by setting the invert elevation drop equal to the value derived from the following equation:

$$\text{invert drop} = (d_u + v_u^2/2g)$$
$$- (d_L + v_L^2/2g) + \Delta E$$

in which,

d_u = depth of flow in incoming sewer (ft)

d_L = depth of flow in outgoing sewer (ft)

Negative invert drops are made equal to zero.

The foregoing does not apply where drop manholes are involved.

For both full flow analysis and partial flow analysis, outlet branch sewers must be located not lower than the incoming main sewer.

4.13 COST DATA

Baffa (3), in a U.S. study, showed that 85% of the cost of gravity flow sewer systems is due to pipe supply and installation costs, and 15% is due to manhole costs.

Cost data was obtained from the Manual of Commercial Estimating and Engineering Standards (25).

The costs were developed from abstracts of unit price bids in the United States.

The costs given are complete in place including all items of work "except shoring of excavations, restore existing improvements, and restore existing permanent surfacing."

The costs given are based on construction "in streets with medium traffic, medium overhead and underground obstructions, light clearing, remove existing 4" asphaltic concrete paving, stable soil conditions, no ground water, haulaway distance of three miles, imported bedding and placing and consolidating backfill."

The reference supplies an array of reinforced concrete main line storm drain costs in U.S. dollars per lineal foot, in pipe sizes from 24 inches to 144 inches and for depths to subgrade of 6, 7, 8, 9, 10, 11, 12, 14, 16, 18, 20, 22, 24, 26 and 28 feet. The costs are shown in Tables 4.2 and 4.3.

As costs were required for pipe sizes down to 8 inch diameter costs were extrapolated. The extrapolation was based on the cost data for diameters

REINFORCED CONCRETE MAINLINE STORM DRAINS	VOLUME C 1 ACCOUNT 2-82 PAGE 3 ISSUE 3

Cost; Dollars Per Lineal Foot for Reinforced Concrete Mainline Storm Drains.

Pipe Size In.	Depth to Subgrade - Feet						
	6'-0"	7'-0"	8'-0"	9'-0"	10'-0"	11'-0"	12'-0"
24"	$17.00	$17.00	$18.00	$18.00	$18.00	$ 19.00	$ 20.00
27"	18.00	18.00	19.00	19.00	19.00	20.00	20.00
30"	19.00	19.00	20.00	20.00	21.00	21.00	22.00
33"	20.00	21.00	21.00	22.00	23.00	24.00	24.00
36"	22.00	23.00	24.00	24.00	25.00	26.00	26.00
39"	25.00	25.00	26.00	27.00	27.00	28.00	29.00
[illegible]	27.00	28.00	29.00	29.00	30.00	31.00	32.00
45"	[illegible]	30.00	31.00	32.00	32.00	33.00	34.00
48"	32.00	33.00	34.00	34.00	35.00	36.00	37.00
51"	35.00	35.00	36.00	37.00	38.00	39.00	40.00
54"	37.00	38.00	39.00	40.00	41.00	42.00	43.00
57"	40.00	40.00	42.00	42.00	43.00	[illegible]	45.00
60"	42.00	43.00	45.00	45.00	46.00	47.00	48.00
63"		46.00	47.00	48.00	50.00	51.00	52.00
66"		49.00	50.00	51.00	53.00	54.00	55.00
69"		53.00	54.00	56.00	57.00	57.00	59.00
72"			57.00	58.00	59.00	60.00	61.00
75"			60.00	61.00	62.00	64.00	65.00
78"			64.00	65.00	66.00	67.00	69.00
81"			67.00	68.00	70.00	71.00	73.00
84"				73.00	74.00	76.00	77.00
87"				77.00	79.00	81.00	83.00
90"				82.00	83.00	85.00	87.00
93"					88.00	90.00	93.00
96"					94.00	95.00	97.00
102"						101.00	104.00
108"						107.00	109.00
114"							114.00
120"							120.00

See Account 2-87 for Drawings. (Main Drawings only. Reference Drawings not included.)

See General Notes, Account 2-80.

(Continued on following page)

Table 4.2 SEWER PIPE COST DATA - Part 1

VOLUME 1	REINFORCED CONCRETE MAINLINE STORM DRAINS
ACCOUNT 2-82 PAGE 4	
ISSUE 3	

(Continued)

Pipe Size In.	Depth to Subgrade - Feet							
	14'-0"	16'-0"	18'-0"	20'-0"	22'-0"	24'-0"	26'-0"	28'-0"
24"	$ 21.00	$ 22.00	$ 24.00	$ 26.00				
27"	22.00	23.00	25.00	26.00	$ 28.00			
30"	24.00	25.00	27.00	28.00	29.00	$ 31.00		
33"	26.00	27.00	29.00	30.00	33.00	35.00	$ 37.00	
36"	28.00	29.00	31.00	33.00	35.00	37.00	40.00	$ 44.00
39"	30.00	32.00	34.00	36.00	39.00	41.00	44.00	48.00
42"	33.00	35.00	37.00	39.00	42.00	44.00	48.00	51.00
45"	36.00	38.00	40.00	42.00	45.00	47.00	51.00	55.00
48"	39.00	41.00	44.00	46.00	48.00	51.00	55.00	59.00
51"	42.00	44.00	46.00	49.00	51.00	54.00	58.00	62.00
54"	46.00	46.00	49.00	53.00	55.00	58.00	62.00	66.00
57"	47.00	51.00	54.00	57.00	59.00	61.00	65.00	70.00
60"	50.00	53.00	56.00	60.00	62.00	65.00	69.00	73.00
63"	54.00	57.00	60.00	63.00	66.00	69.00	73.00	77.00
66"	57.00	60.00	63.00	66.00	69.00	73.00	77.00	81.00
69"	60.00	64.00	67.00	70.00	73.00	76.00	81.00	86.00
72"	64.00	67.00	71.00	74.00	77.00	81.00	86.00	90.00
75"	67.00	71.00	74.00	78.00	82.00	85.00	90.00	95.00
78"	71.00	75.00	79.00	82.00	86.00	90.00	95.00	100.00
81"	76.00	79.00	83.00	87.00	91.00	95.00	100.00	105.00
84"	80.00	83.00	87.00	91.00	95.00	100.00	105.00	111.00
87"	86.00	90.00	92.00	97.00	102.00	107.00	113.00	119.00
90"	91.00	96.00	100.00	105.00	108.00	114.00	120.00	127.00
93"	98.00	102.00	107.00	112.00	117.00	123.00	125.00	135.00
96"	101.00	106.00	117.00	117.00	123.00	129.00	136.00	141.00
102"	108.00	114.00	119.00	125.00	131.00	138.00	[illegible]	154.00
108"	114.00	119.00	123.00	130.00	137.00	145.00	[illegible]	161.00
114"	119.00	122.00	128.00	136.00	143.00	151.00	160.00	165.00
120"	125.00	130.00	136.00	143.00	150.00	158.00	167.00	176.00
126"	134.00	139.00	145.00	151.00	161.00	171.00	180.00	189.00
132"	142.00	149.00	155.00	161.00	172.00	183.00	192.00	201.00
138"	152.00	160.00	167.00	174.00	185.00	195.00	204.00	214.00
144"	163.00	171.00	180.00	189.00	197.00	207.00	217.00	227.00

See Account 2-87 for Drawings. (Main Drawings only. Reference Drawings not included.)

See General Notes, Account 2-80.

Table 4.3 SEWER PIPE COST DATA - Part 2

from 24 inch to 42 inch. The following cost equation was obtained from the aforementioned subset of cost data by using UWHAUS (38), a non-linear least squares computer program utilising Marquardt's method.

UNITP = 9.73 + 0.097* CUT ** 1.6 + .826* DIA ** 2.387

in which,

UNITP = pipe cost per lineal foot ($)

CUT = depth to subgrade (ft)

DIA = diameter of pipe (ft)

The above equation was used to extrapolate the cost data and produce a new cost array which was used in the program. This cost array is shown in table 4.4. In addition graphs of cost vs cut and cost vs diameter were plotted for some values of diameter and cut respectively. These graphs are shown in figures 4.8 and 4.9.

The new costs were produced to the first decimal place; the original costs were to the nearest dollar. The c equation permitted costs to be computed for a continuous range of depths. The new cost data, therefore, permitted costs to be computed over a more continuous range than the original data.

The type of manhole selected for the cost data is shown on figure 4.10. This manhole is designed for sewer sizes less than 36 inch diameter. The cost of this manhole is equal to the cost of the manhole chamber plus the cost of the shaft. Shaft costs are shown in table 4.5.

PIPE COST ARRAY

DIA	DEPTH (FT) 6	7	8	9	10	11	12	14	16	18	20	22	24	26	28
8	11.7	12.2	12.7	13.3	13.9	14.5	15.2	16.7	18.2	19.9	21.8	23.7	25.7	27.9	30.1
10	12.0	12.4	13.0	13.5	14.1	14.8	15.4	16.9	18.5	20.2	22.0	23.9	25.9	28.1	30.3
12	12.3	12.7	13.3	13.8	14.4	15.1	15.7	17.2	18.7	20.4	22.3	24.2	26.2	28.4	30.6
15	12.8	13.3	13.8	14.4	15.0	15.6	16.3	17.8	19.3	21.0	22.8	24.8	26.8	28.9	31.2
18	13.6	14.1	14.6	15.2	15.8	16.4	17.1	18.5	20.1	21.8	23.6	25.5	27.6	29.7	32.0
21	14.6	15.1	15.6	16.1	16.7	17.4	18.0	19.5	21.1	22.8	24.6	26.5	28.5	30.7	32.9
24	15.8	16.2	16.8	17.3	17.9	18.5	19.2	20.7	22.2	23.9	25.8	27.7	29.7	31.9	34.1
27	17.2	17.6	18.2	18.7	19.3	20.0	20.6	22.1	23.6	25.3	27.2	29.1	31.1	33.3	35.5
30	18.8	19.3	19.8	20.4	21.0	21.6	22.3	23.7	25.3	27.0	28.8	30.7	32.8	34.9	37.1
33	20.7	21.2	21.7	22.2	22.8	23.5	24.1	25.6	27.2	28.9	30.7	32.6	34.6	36.8	39.0
36	22.8	23.3	23.8	24.4	25.0	25.6	26.3	27.7	29.3	31.0	32.8	34.7	36.8	38.9	41.2
39	25.2	25.7	26.2	26.8	27.4	28.0	28.7	30.1	31.7	33.4	35.2	37.1	39.2	41.3	43.6
42	27.9	28.3	28.9	29.4	30.0	30.7	31.3	32.8	34.4	36.1	37.9	39.8	41.8	44.0	46.2

Table 4.4 EXTRAPOLATED SEWER PIPE COST DATA

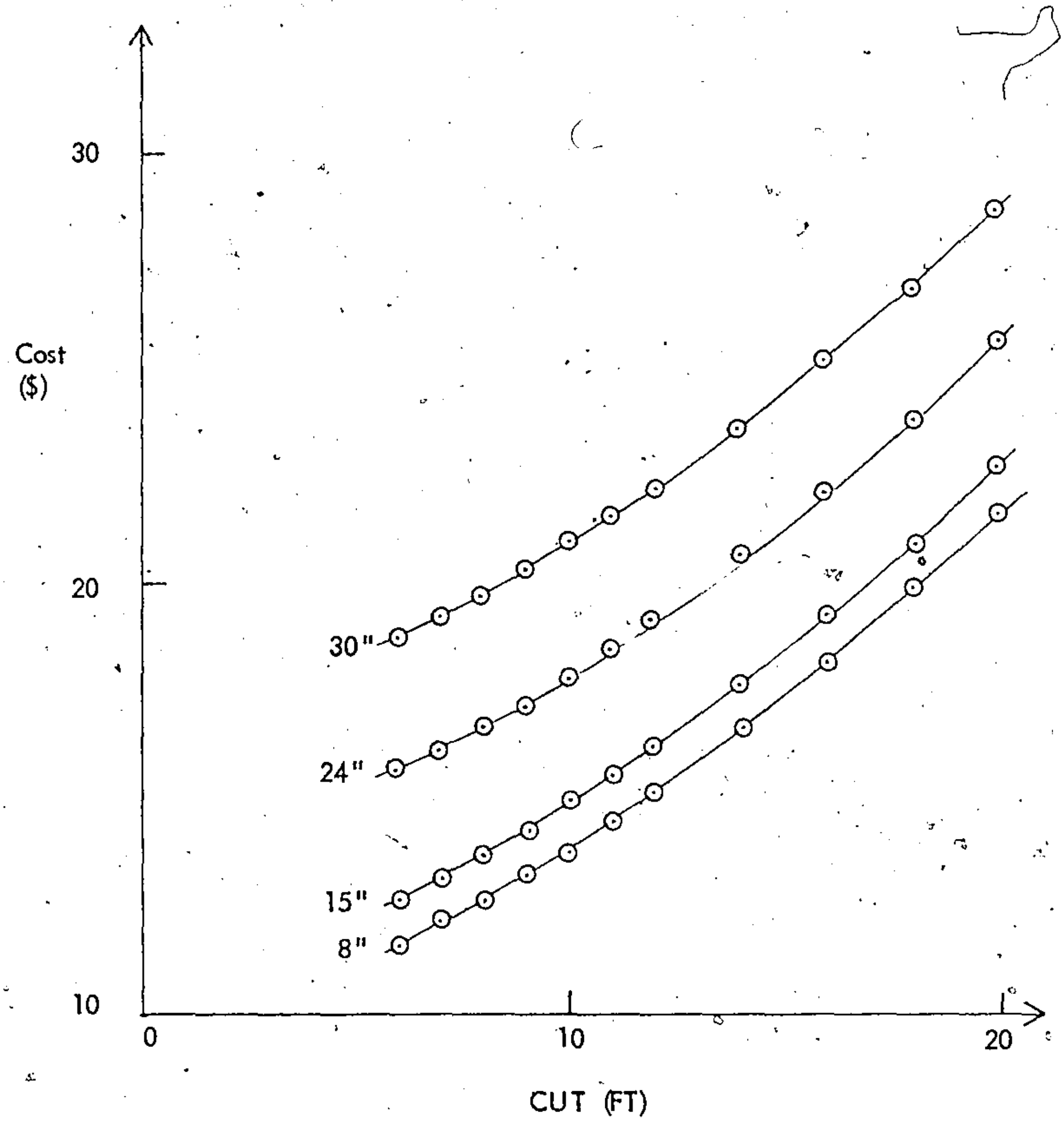


Figure 4.8 PIPE COST vs CUT

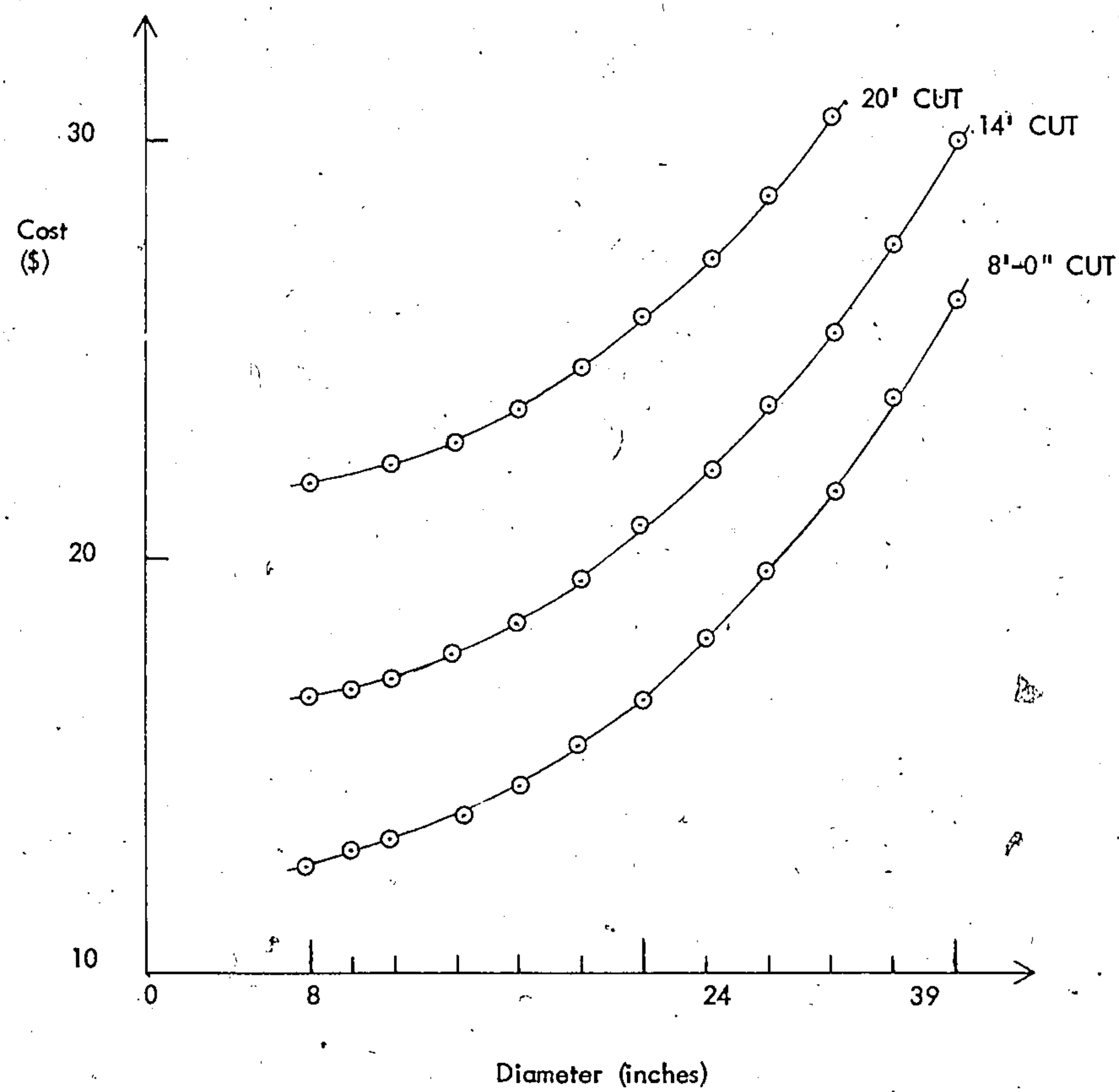


Figure 4.9 PIPE COST vs DIAMETER

STORM DRAIN MANHOLES	VOLUME 1 ACCOUNT 2-83 PAGE 1 [illegible] 1

Following gives current standard costs for Manholes complete in place in accordance with Standard Drawings Account 2-87.

The costs are complete in place including all items of work except shoring of excavations if required, restore existing improvements and restore existing permanent surfacing.

Manhole Number 1, Drawing R-136, Account 2-87

For total cost use $303.00 plus cost of shaft as follows:

Shaft Height Feet	Cost Each
1	$154.00
2	198.00
3	248.00
4	275.00
5	308.00
6	341.00
7	385.00
8	418.00
9	462.00
10	506.00
11	539.00
12	572.00
13	605.00
14	644.00
15	682.00
16	715.00
17	743.00
18	770.00
19	814.00
20	836.00
21	869.00
22	897.00
23	924.00
24	957.00
25	979.00

Example: Manhole Number 1 with Shaft Height of 10'0"

Manhole = $303.00
Shaft = $506.00
Total = $809.00

See General Notes, Account 2-80.

(Continued on following page)

Table 4.5 SHAFT COST DATA

COMMERCIAL - INDUSTRIAL ESTIMATING & ENGINEERING STANDARDS

MANHOLE NO. 1 PLAN, SECTION AND DETAIL DRAWING R-136	VOLUME I ACCOUNT 2-87 PAGE 21 ISSUE 3

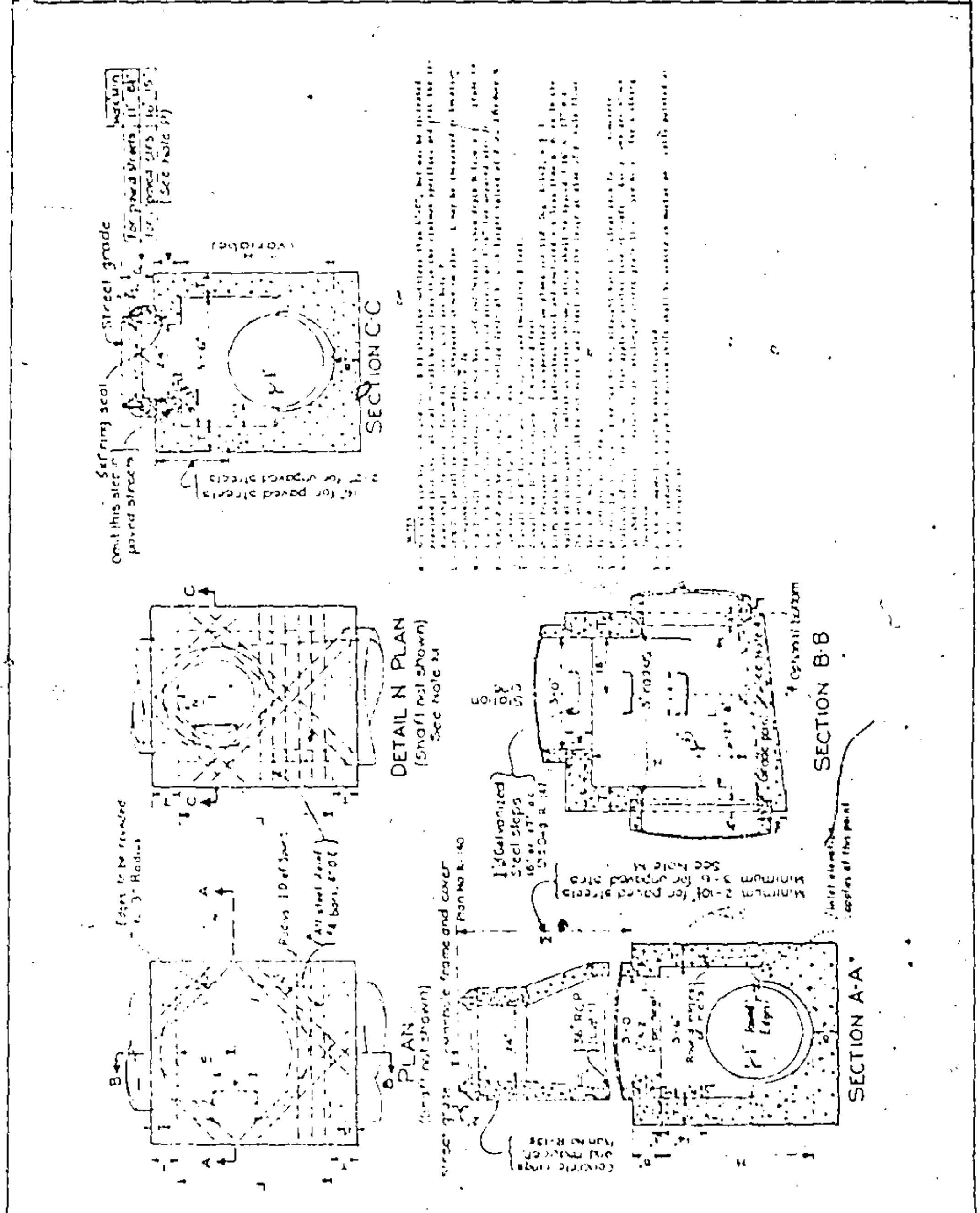


Figure 4.10 TYPICAL MANHOLE DETAILS

To obtain the shaft costs as a continuous function of the shaft height, the shaft cost data of table 4.5 was used in a least squares polynomial curve fitting computer program (20) to obtain the following equation:

PRICES = 121.18 + 37.81 * HEIGHT + .089 * HEIGHT **2 - .009 * HEIGHT ** 3

in which,

PRICES = shaft cost for height HEIGHT. ($)

HEIGHT = height of shaft (ft)

This equation is a smooth curve which is almost a straight line relationship for heights greater than 2 feet.

Drop pipe costs were taken to be given by:

UNITP = 10.+ 10.* DIA

in which,

UNITP = cost per lineal foot of the drop pipe ($)

DIA = diameter of pipe (ft)

4.14 INPUT DATA

The configuration of the network which the program will handle is a function of the number of sewers, junctions and the available fall across the system.

Data preparation and entry is described below.

Program Sewer

Program SEWER allows the user to set the dimensions for subroutine DESIGN by entering the values of LINES, NJMAX and NSMAX. This adjustable dimension feature permits the user to use just sufficient computer storage for the problems to be run.

LINES is the number of sewers in the network plus 1. The maximum value of LINES, for the batch of problems to be run, must be entered.

NJMAX is the maximum value of the number of junctions in the batch of problems to be run. The minimum value of NJMAX is 1.

NSMAX is the maximum value of the number of states in the batch of problems to be run. The number of states is the available fall in feet across the system divided by 0.1, and minus 25.

Function PRICEP (HEIGHT, DIA)

The user must enter an equation, with diameter and cut as variables, to compute the supply and installation cost of the pipe per lineal foot.

Function PRICED (HEIGHT, DIA)

The user must enter an equation to compute the drop pipe cost per lineal foot.

Function PRICES (HEIGHT)

The user must enter an equation to compute the manhole shaft cost.

Full details of the format of PRICEP, PRICED and PRICES can be seen in the program listing (Appendix 1).

The input data card requirements are described below. All elevations and cover requirements must be to the first decimal place.

Data Cards

1. Card type 1 contains the date and name of the user in the format. (I2, A3, I4, 5A4) e.g.

   20APR1975 BILL MAIN

2. Card type 2 contains the number of problems to be run NPROB, and the value of ICOST in the format (2I2). If ICOST = 0, the cost array data is used and card types 3, 4 and 5 are required. If ICOST = 1, the cost functions are used to compute costs and card type 6 is required.

3. Card type 3 contains the pipe size IDIA, and supply and installation costs CPLF for each value of cut in the format (I4, 15F5.1). One card is required for each pipe size.

4. Card type 4 contains SCLF the shaft cost array data in the format (8F10.5).

5. Card type 5 contains DCLF the drop pipe cost array data in the format (8F10.5).

6. Card type 6 contains DIA the pipe sizes considered for the problem in the format (8F10.5).

7. Card type 7 contains the manhole box cost CBOX in the format (F10.5).

8. Card type 8 contains the problem name IREM in the format (12A4).

9. Card type 9 contains YES in format (A3) if there is to be a change in any of the design criteria listed in the program. Otherwise, the card is left blank.

10. Card type 10 is required only if there is a design criteria change. The following data must be entered in a (8F10.5) format in the order listed:

    (i) smallest allowable diameter DIAMIN (inches),

    (ii) Manning's roughness coefficient, RMANN,

    (iii) minimum full flow velocity, VMIN (f.p.s.),

    (iv) maximum velocity, VMAX (f.p.s.),

    (v) minimum cover, COVMIN (fr),

    (vi) maximum cover, COVMAX (ft),

(vii) value of PEAKFS. If PEAKFS = Ø, PKF = $(18 + \sqrt{P})/(4 + \sqrt{P})$

If PEAKFS = 1, PKF = $5/P^{0.2}$

(viii) value of SIGN3. SIGN3 = 0 for full flow analysis.

SIGN3 = 3 for partial flow analysis.

11. Card type 11 contains YES in an (A3) format if there is to be any change in the flow criteria listed in the program. Otherwise the card is left blank.

12. Card type 12 is required only if there is a flow criteria change. The following must be entered in the order listed in a (4F10.5) format - domestic sewage flow in Imperial gallons per capita per day; commerical sewage, industrial sewage and infiltration flow rates in Imperial gallons per acre per day.

13. Card type 13 contains the maximum permissible outfall diameter OUTDIA (inches) and the minimum permissible outfall energy line elevation OUTFAL (ft) in the format (2F10.5). The maximum permissible diameter cannot exceed 39 inches for the cost array data option.

14. One card type 14 is required for each sewer in the network. The card contains the following information in a (11F7.3) format in the order listed:

(i) upstream manhole number UMHN,

(ii) downstream manhole number DMHN,

(iii) upstream ground elevation, UGELEV (ft),

(iv) downstream ground elevation, DGELEV (ft),

(v) length of sewer RUN (ft),

(vi) incremental acreage tributary to the sewer, ACRES,

(vii) population density, POPD (persons per acre),

(viii) type of area, TYPE, e.g.

domestic, TYPE = 1

commercial, TYPE = 2

industrial, TYPE = 3,

(ix) additional flow, QADD (c.f.s.)

(x) upper elevation constraint at the downstream manhole, ELCU (ft),

(xi) lower elevation constraint at the downstream manhole, ELCL (ft).

15. Card type 15 is a blank card to signal the end of the collection system data and to assign the number of sewers NLINES in the system.

The data card sequence for a problem is shown in figure 4.11.

<u>Test Problem 1</u>

The program was tested and debugged using the problem shown in figure 4.1. This problem was adapted from Rich (34). An example of input data for this problem is shown in figure 4.12.

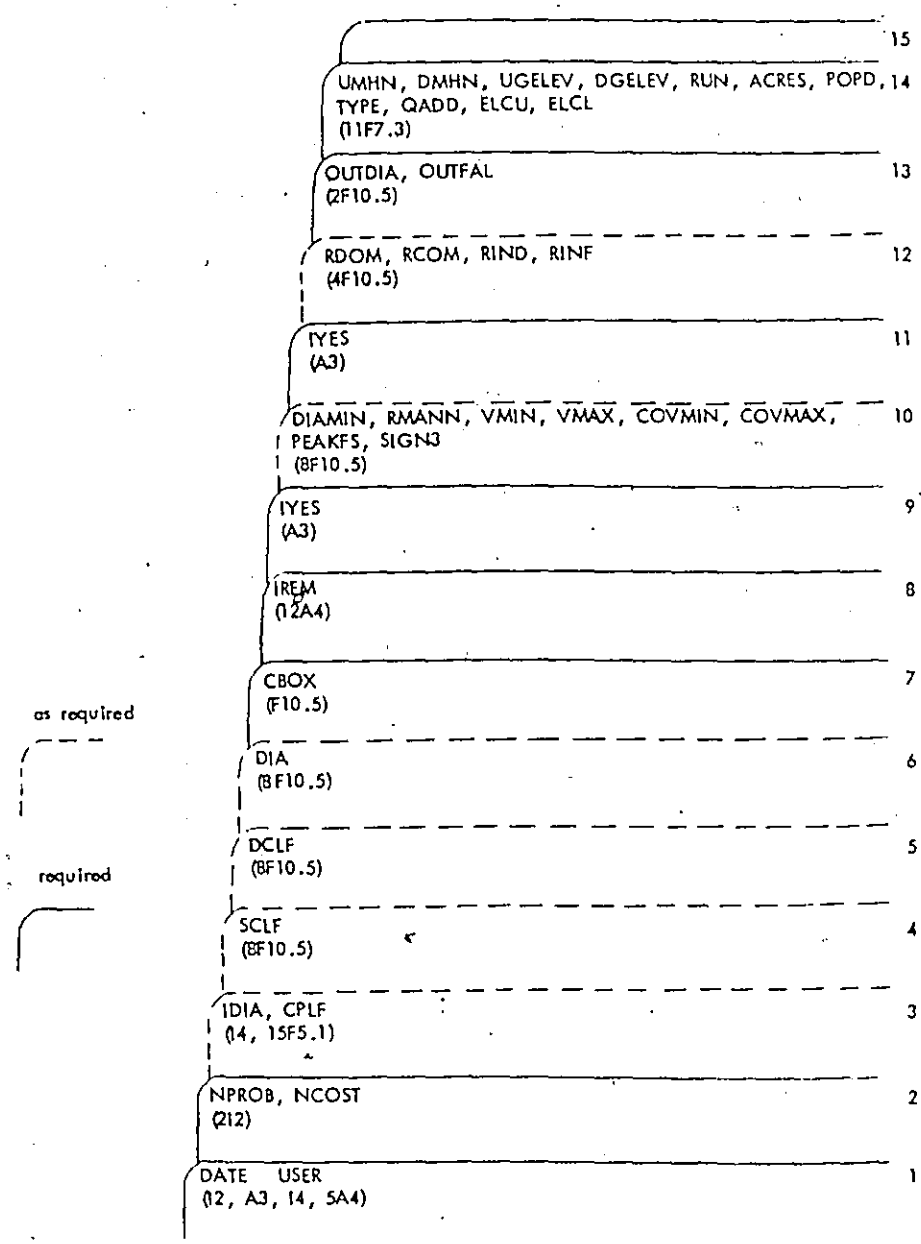


Fig. 4.11 DATA CARD ARRANGEMENT

```
18JUN1975 BILL MAIN
 1 1
   8.       9.        12.        15.        18.        21.        24.        27.
  30.      33.        36.        39.        42.
 303.
TEST PROBLEM 1
YES
   8.        .013      2.        10.         8.        15.0                  3.0
YES
 106.                           1000.
  24.        204.0
  1.6     1.5  228.2  224.2  400.       2.5   55.      1.
  1.5     1.4  224.2  222.2  400.       2.5   55.      1.
  1.4     1.3  222.2  219.0  400.       2.5   55.      1.
  5.1     1.3  229.0  219.0  400.       2.5   55.      1.
  1.3     1.2  219.0  215.4  300.       0.     0.      1.
  4.1     1.2  224.4  215.4  400.       2.5   55.      1.
  2.3     2.2  225.4  221.6  400.       2.5   55.      1.
  3.1     2.2  222.8  221.8  100.       0.     0.      1.        3.09
  2.2     2.1  221.8  220.6  400.       2.5   55.      1.
  2.1     1.2  220.6  215.4  400.       2.5   55.      1.
  1.2     1.1  215.4  215.00 300.       0.     0.      1.
```

Figure 4.12 DATA CARD INPUT FOR TEST PROBLEM 1

4.15 OUTPUT DATA

The program prints out the information used in the design of the sewer system - cost array data if used, manhole box cost, pipe sizes considered, design and flow criteria, and the sewer system information. The output listing showing this for test problem 1 is shown in figure 4.13.

If the program is executed successfully, the program will print out, for each pass, the optimum sewer sizes, vertical alignment data, sewer costs and cover over the sewers. The second pass output listing showing this for test problem 1 is given in figure 4.14. The variable names in the listing, except those below, are given in appendix 2.

QMAX = maximum design flow

QMIN = minimum design flow

MANH LOSS = energy loss at manhole

UINV ELEV = upstream invert elevation

DINV ELEV = downstream invert elevation

The program also prints out the following information

1. maximum design flow in each sewer,
2. best cumulated cost and decision for each state and each stage (tied solutions are marked by an asterisk),
3. best pipe sizes for the stage,
4. optimum decisions for the sewer network,

```
MANHOLE BOX COST = 333.0

PIPE DIAMETERS (INCHES) CONSIDERED =
   8.   10.   12.   15.   18.   21.   24.   27.   30.   33.
  36.   39.   42.
```

```
SANITARY SEWER DESIGN BY DYNAMIC PROGRAMMING
RUN  1      DATE 18 JUN  1975       USER BILL MAIN


TEST PROBLEM 1


DESIGN CRITERIA


MINIMUM DIAMETER (INCHES) =   8.
MANNINGS N = .013
MINIMUM VELOCITY (FPS) =  2
MAXIMUM VELOCITY (FPS) = 10
MINIMUM COVER (FT) = 8.0
MAXIMUM COVER (FT) =15.0
PEAKING FACTOR = HARMON FORMULA
NON FULL FLOW CONDITIONS CONSIDERED
DOMESTIC FLOW (IMP GALS/CAP/DAY) =136.
COMMERCIAL FLOW (IMP GALS/ACRE/DAY) =      -0.
INDUSTRIAL FLOW (IMP GALS/ACRE/DAY) =      -0.
INFILTRATION (IMP GALS/ACRE/DAY) =  1000.


COLLECTION SYSTEM DATA


OUTFALL DIAMETER (INCHES) = 24.
OUTFALL ENERGY LINE ELEVATION =204.00
```

LINE	UMHN	DMHN	UGELEV (FT)	DGELEV (FT)	RUN (FT)	ACRES	POPD	TYPE	QADD (CFS)
1	1.600	1.500	228.20	224.20	400.	2.5	55.	1.	-0.00
2	1.500	1.400	224.20	222.20	400.	2.5	55.	1.	-0.00
3	1.400	1.300	222.20	219.00	400.	2.5	55.	1.	-0.00
4	5.100	1.300	229.00	219.00	400.	2.5	55.	1.	-0.00
5	1.300	1.200	219.00	215.40	300.	0.0	0.	1.	-0.00
6	4.100	1.200	224.40	215.40	400.	2.5	55.	1.	-0.00
7	2.300	2.200	225.40	221.80	400.	2.5	55.	1.	-0.00
8	3.100	2.200	222.60	221.80	100.	0.0	0.	1.	3.[illegible]9
9	2.200	2.100	221.80	220.60	400.	2.5	55.	1.	-0.00
10	2.100	1.200	220.60	215.40	400.	2.5	55.	1.	-0.00
11	1.200	1.100	215.40	215.00	300.	0.0	0.	1.	-0.00

```
DOMESTIC = 1.      COMMERCIAL = 2.      INDUSTRIAL = 3.
```

Figure 4.13 OUTPUT LISTING PART 1 – TEST PROBLEM 1

LINE	QMAX (CFS)	QMIN (CFS)	SLOPE	DIAM (IN)	QFULL (CFS)	VFULL (FPS)	VMAX (FPS)	VMIN (FPS)	MAX DEPTH (FT)
1	.12	.03	.0100	8.0	1.21	3.5	1.9	1.6	.16
2	.23	.06	.0048	8.0	.83	2.4	1.7	1.4	.27
3	.34	.10	.0080	8.0	1.08	3.1	2.3	1.9	.28
4	.12	.03	.0250	8.0	1.91	5.5	2.7	2.3	.14
5	.47	.13	.0120	8.0	1.32	3.6	2.9	2.4	.30
6	.12	.03	.0225	8.0	1.81	5.2	2.6	2.2	.14
7	.12	.03	.0090	8.0	1.15	3.3	1.8	1.5	.17
8	3.09	3.09	.0100	12.0	3.56	4.5	4.6	4.6	.80
9	3.32	3.15	.0030	15.0	3.54	2.9	3.0	2.9	1.06
10	3.43	3.19	.0125	15.0	7.22	5.9	4.9	5.3	.67
11	4.00	3.34	.0063	15.0	5.14	4.2	4.1	4.0	.93

LINE	MAX ENERGY (FT)	MANH LOSS (FT)	INVERT DROP (FT)	FALL (FT)	UINV ELEV (FT)	DINV ELEV (FT)	UGELEV (FT)	OGELEV (FT)
1	.22	0.00	0.00	4.00	219.41	215.41	228.20	224.20
2	.31	.02	.11	1.90	215.30	213.42	224.20	222.20
3	.36	.02	.07	3.20	213.34	210.16	222.20	219.00
4	.26	0.00	0.00	10.00	220.27	210.27	229.00	219.00
5	.43	.02	.09	3.60	210.07	206.49	219.00	215.40
6	.25	0.00	0.00	9.00	215.68	206.68	224.40	215.40
7	.22	0.00	0.00	3.60	216.61	213.01	225.40	221.80
8	1.13	0.00	0.00	1.00	213.69	212.69	222.80	221.80
9	1.20	.02	1.00	1.20	212.01	210.83	221.80	220.60
10	1.04	.05	0.00	5.00	210.83	205.99	220.60	215.40
11	1.19	.03	.79	1.90	205.71	203.83	215.40	215.00

LINE	COST
1	6532.98
2	5959.40
3	5977.78
4	5944.53
5	4654.42
6	5942.86
7	6532.48
8	2009.69
9	6676.28
10	6001.91
11	4683.05

TOTAL COST = 60915.36 DOLLARS

UMHN	COVER (FT)
1.60	8.1
1.50	8.2
1.40	8.1
5.10	8.0
1.30	8.2
4.10	8.0
2.30	8.1
3.10	8.0
2.20	8.4
2.10	8.4
1.20	8.3

Figure 4.14 OUTPUT LISTING PART 2 - TEST PROBLEM 1

5. energy drop, if required,

6. energy line elevation at the downstream end of each sewer.

Items (2) to (6) are printed out for each pass. The program may be terminated at the end of the first pass by assigning the initial value of SIGN4 = 4 in the program deck.

If the problem is not executed successfully, a message is printed to show that an error has occurred.

The messages are:

1. "number of states too large"

2. "job terminated at manhole number = **, insufficient fall"

3. "job terminated at manhole number = **, outfall diameter is too small"

4. "job terminated at outfall manhole"

5. "Error on branch = ** at stage = **"

6. "Error at dummy sewer assignment on branch = **"

CHAPTER 5.0 DISCUSSION OF PROGRAM

This chapter contains an analysis of the method of functioning of the program and a comparison with other programs.

5.1 CORRECTNESS OF PROGRAM

The correctness of the program is the ability of the program to produce a realistic optimum solution. The correctness is discussed under the following headings:

1. Treatment of Branches,
2. Hydraulic Mathematical Computation Methods,
3. Vertical Alignment at Manholes,
4. Cost Data.

1. Treatment of Branches

There is a remote possibility that the method used to accumulate branch costs may not result in an optimal solution being obtained. Prior to accumulating a branch cost to the sewer system cost, the branch costs are sorted to obtain the cheapest cost for each state. This is shown in the hypothetical example in figure 5.1. After sorting, it will be noted that the sewer size for the lowest state has been increased from 8 inch to 12 inch diameter, thereby restricting the sewer sizes downstream and contributory to this state to a minimum 12 inch size. This requirement may not result in the cheapest sewer system cost. If the branch cost was sorted to produce the

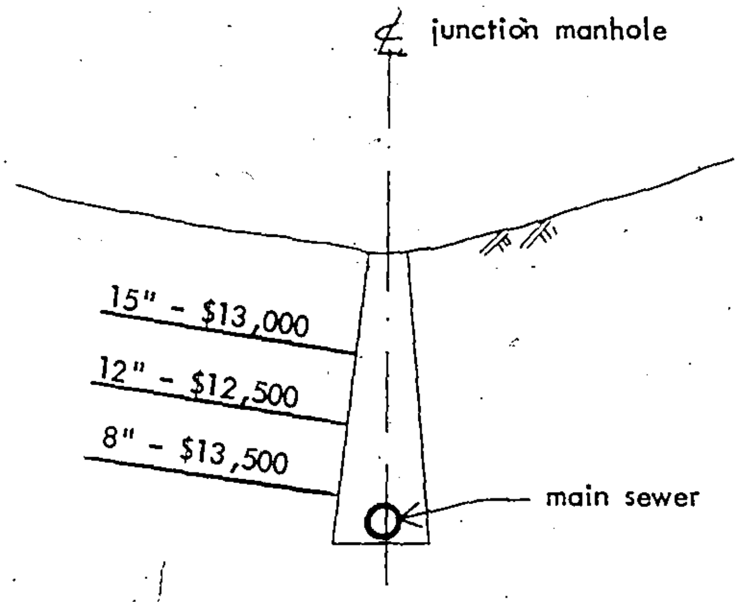


Diagram showing hypothetical sewer sizes and costs before sorting.

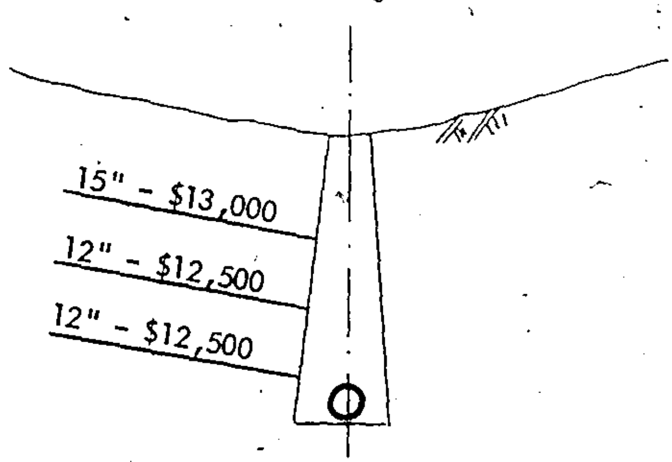


Diagram showing hypothetical sewer sizes and costs after sorting.

Figure 5.1 DIAGRAM ILLUSTRATING BRANCH SORTING PROCEDURE smallest diameter from each state, then a cheaper sewer system cost may be obtained. However, in most situations, the method adopted is the more practical because the non-decreasing diameter requirement is not critical as branch outlet sewers are generally smaller in size than the main sewer.

2. Hydraulic Mathematical Computational Methods

Subroutines (35) were available to compute hydraulic elements analytically. However semi-graphical methods were used as they were computationally more efficient.

It is believed that no appreciable error is incurred with respect to the hydraulic computations. Some error occurs in the polynomial relationships derived from the hydraulic-elements charts. The accuracy of these polynomial equations is a function of the accuracy of interpolation and the number of data points used to derive these equations. However, the error is not considered to be appreciable or possibly any greater than would occur with hand calculations. In addition, it may be noted that the curves for n variable with depth were used. These curves were based on the average of 824 experiments. Design and Construction of Sanitary and Storm Sewers (12) states that the n variable curves may be questioned because of differing results from various researchers and the decision to use n variable with depth of flow must be left to the individual designer.

Pomeroy's (32) formula is used to compute the velocity in the sewer for the minimum design flow QDMIN. Error may occur here because a linear relationship between Manning's roughness coefficient and Pomeroy's constant K was obtained using only three data points. In addition, Pomeroy's formula was developed from data on sewers not greater than 24 inches diameter. A computer program was written to compare the velocities obtained by Pomeroy's equation for pipe sizes up to 42 inch diameter at various slopes and relative flow depths. Manning's equation was solved by successive approximations using functions PIPDUF, PIPROP and NORMLQ from Smith (35). The results obtained with the 24 inch and 36 inch diameter pipes are shown in table 5.1.

The results from both methods of computation are close. As the minimum velocity is not a criterion in the selection of sewer size, the computation of the minimum velocity is not critical.

3. Vertical Alignment at Manholes

The method of computation of the vertical alignment of the sewers at manholes in partial flow analysis results in some error. No allowance is made for the additional energy loss at junction manholes. In addition, the allowance for the energy loss at the manhole means that the sewer is not located parallel to the energy line. This is illustrated in figure 5.2. It was not considered practical to correct for these errors, because energy losses at manholes are of the magnitude of 0.02 foot and the energy line displacement would not appreciably change the design of the sewer.

4. Cost Data

The cost data are the most important part of the program. It is essential that they be realistic and up-to-date if the program is to be useful.

DIAMETER (FT) = 2.00 N = 0.013 VFULL (FPS) = 2.0
SLOPE = 0.0008

QRATIO	VELOCITY (FPS) MANNING	VELOCITY (FPS) POMEROY
0.10	1.08	1.14
0.20	1.31	1.34
0.30	1.47	1.48
0.40	1.61	1.59
0.50	1.72	1.68
0.60	1.80	1.75
0.70	1.89	1.82
0.80	1.97	1.86
0.90	2.04	1.93

DIAMETER (FT) = 2.00 N = 0.013 VFULL (FPS) = 10.0
SLOPE = 0.0191

QRATIO	VELOCITY (FPS) MANNING	VELOCITY (FPS) POMEROY
0.10	5.15	6.09
0.20	6.30	7.19
0.30	7.07	7.92
0.40	7.70	8.49
0.50	8.28	8.95
0.60	8.72	9.35
0.70	9.15	9.71
0.80	9.47	10.02
0.90	9.84	10.31

DIAMETER (FT) = 3.00 N = 0.013 VFULL (FPS) = 2.0
SLOPE = 0.0004

QRATIO	VELOCITY (FPS) MANNING	VELOCITY (FPS) POMEROY
0.10	1.08	1.11
0.20	1.31	1.31
0.30	1.47	1.44
0.40	1.61	1.55
0.50	1.72	1.63
0.60	1.80	1.70
0.70	1.89	1.77
0.80	1.97	1.83
0.90	2.04	1.86

DIAMETER (FT) = 3.00 N = 0.013 VFULL (FPS) = 10.0
SLOPE = 0.0111

QRATIO	VELOCITY (FPS) MANNING	VELOCITY (FPS) POMEROY
0.10	5.15	5.92
0.20	6.30	7.00
0.30	7.07	7.71
0.40	7.70	8.26
0.50	8.28	8.72
0.60	8.72	9.11
0.70	9.15	9.45
0.80	9.47	9.76
0.90	9.84	10.04

Table 5.1 COMPARISON OF VELOCITIES COMPUTED BY MANNING'S AND POMEROY'S EQUATIONS

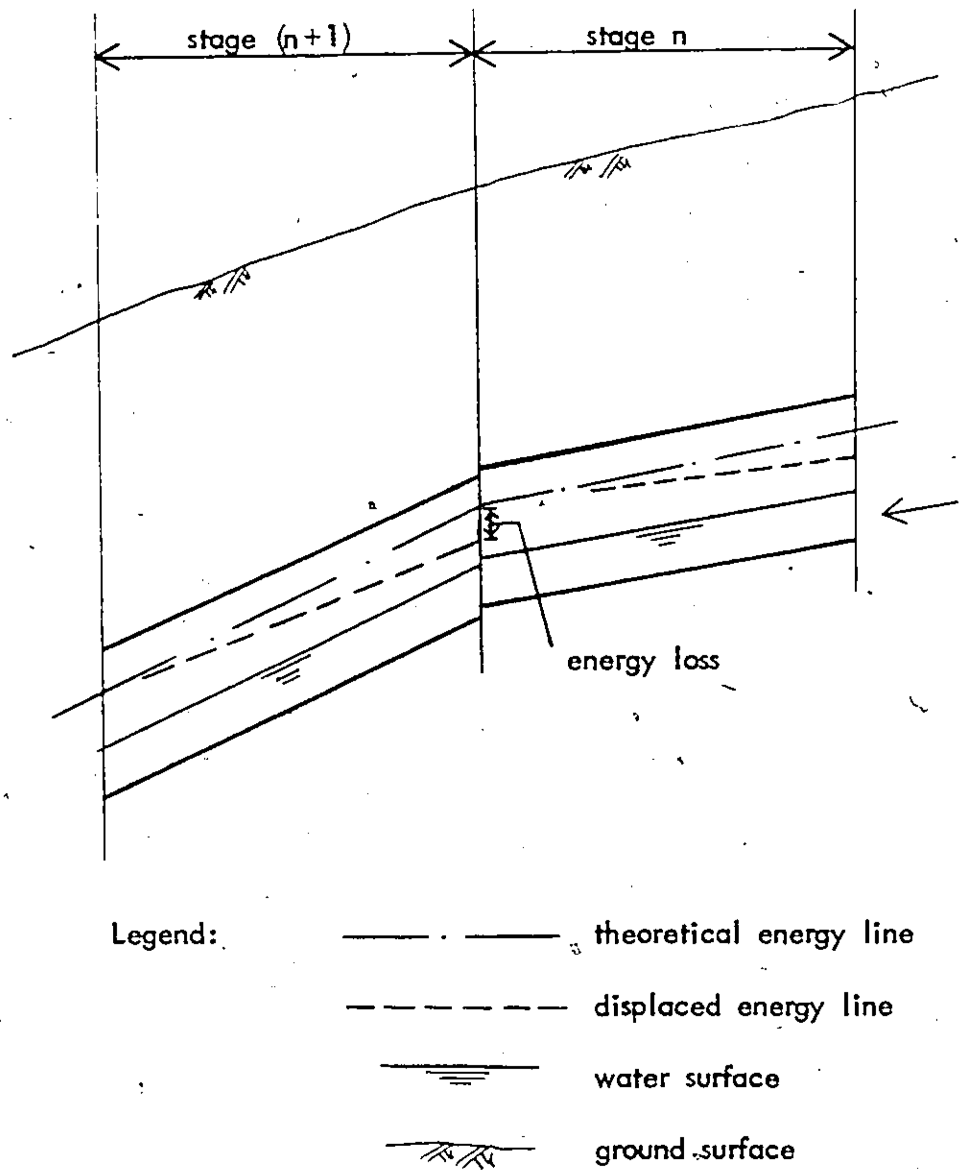


Figure 5.2 DIAGRAM SHOWING DISPLACEMENT OF THEORETICAL ENERGY LINE

A comparison of bids for any sewer project reveals wide variations in bid prices. A contractor's bid reflects his bidding strategy, expertise in the type of work, the location of the job, his staff and equipment availability and the general economic climate. It is therefore impossible to obtain unit prices which one can confidently state are correct. All one can hope to state is that they are the best available.

Cost data can be obtained by approaching sewer contractors or analysing past contracts. Inviting contractors to examine proposed plans and advise of the expected costs is probably the best way of obtaining realistic costs.

Analysis of past contract prices may be performed by the designer or by using outside cost consultants. The author used cost data provided by Richardson's Engineering Services, Inc. (25). Their estimating manual is a well known and comprehensive manual used for estimating civil engineering works.

The author believes the costs are superior to that obtainable from Englesman (13), Yardsticks for Costing (40) or by analysing local data, taking into consideration the purposes of the program. The estimating manual provided manhole costs, and supply and installation costs for sewer sizes down to a minimum of 24 inch diameter. As described in section 4.13, a cost function was obtained from this data by means of a non-linear estimation computer model. This cost function was used to compute a new cost array for pipe sizes from 8 inches to 42 inches diameter.

The author believes that the pipe and manhole costs are homogeneous and provide a satisfactory cost basis for the purposes of the program.

5.2 STORAGE REQUIREMENTS

The storage requirement of the program is dependent on the number of sewers and configuration of the network. Because of the large storage requirement all the best decisions must be stored on a scratch tape or file. For a problem containing 15 sewers, 5 junctions and an available fall across the system of 17.4 ft., the program requires 64,000 words of central memory on a CDC 6400 computer.

To eliminate wasted storage and to provide flexibility with regard to the configuration of the networks to be solved, an adjustable dimension feature was included in the program.

The main factor in setting the storage requirements is the value of the state increment. The minimum state increment was set at 0.1 ft. If the minimum state increment was increased to 0.2 ft., problems with about twice the available fall across the system could be handled.

Considerable storage is utilised in storing branch outlet costs and pipe sizes two 3-dimensional arrays are required for each junction manhole. In a real life situation normally that branch solution which yields the cheapest cost is added to the system. If only this cheapest cost solution was stored, considerable savings would be made in the storage requirements.

The method of storing the design data in the program could be more efficient because only a small portion of the appropriate arrays are used for each stage. For example, if the available fall across a 3 stage system is

21 units and this fall is divided equally between the 3 stages, then only a 7 unit portion of the array would be used at each stage. It would be more economical to use a 7 unit array and relate the adjacent arrays by the minimum state for each stage, say.

5.3 COARSE GRID APPROACH

The program uses a two pass system in order to obtain some execution time savings. It seems reasonable to assume that the program should produce the vertical alignment data to an accuracy of 0.1 foot. This requires that the states and decisions be incremented in 0.1 foot units. To examine all the feasible states at 0.1 foot increments would be prohibitive. Therefore the program uses 1 foot increments in the first pass and 0.1 foot increments in the second pass. In the first pass, the program searches through the entire feasible region for an optimum solution. The first pass solution defines the feasible region to be searched in the second pass. The feasible region has its upper and lower limits a distance of Y above and below the optimum states obtained in the first pass, providing there are no limiting constraints.

The computational advantage of the two pass system can be shown with reference to a 3-stage problem in which the depth of the feasible region is 9 ft. and Y = 0.5 ft. With a state increment of 0.1 ft. over the entire feasible region, the return function would have to be obtained for 273 states; the number of additions and calculations would be 4004. With the two pass solution the return function would have to be calculated for 63 states; the number of additions and subtractions would be 442.

An increment of 1 foot was selected in the first pass because this seems a reasonable value and the cost array data is based on a 1 foot minimum increment.

The value of Y to ensure that an optimum solution is reached is dependent on the continuity of the cost data, the expected difference between the first and second solution and the first pass increment. If the cost array data is used, a value of Y = 2 feet is recommended as 2 feet is the maximum cost array increment. If the cost equation data is used, a value of Y = 0.5 ft. is the minimum recommended. The value of Y is probably best determined from experience.

The strategy of a two pass solution is totally dependent on the return functions at each stage being unimodal. The pipe cost was shown to be unimodal in the following way. A hypothetical 350 ft. long sewer (n = .013) located in a flat ground area, was designed to carry a flow of 5 c.f.s. The minimum depth to subgrade was set at 9 ft. The minimum fall across the sewer was set at 1 ft. and increased in 1 ft. increments. For each slope, the theoretical pipe size for full flow was calculated by Manning's equation and the unit cost was computed from the pipe cost equation. The results are shown in table 5.2.

Table 5.2 Pipe Diameter vs Cost

FALL (ft)	Pipe Diameter (ft)	Cost per Lineal Foot ($)
1	1.43	16.22
2	1.26	16.01
3	1.17	16.00
4	1.11	16.07
5	1.06	16.17
6	1.03	16.30
7	1.00	16.45
8	0.97	16.62
9	0.95	16.80
10	0.93	17.00
11	0.92	17.21
12	0.90	17.43

5.4 COMPARISON WITH OTHER PROGRAMS

It is difficult to make a comparison with other programs because not all the researchers provide sufficient information to compare details. It is difficult to assess computational efficiency because this depends on the computer processor and one has to know the area of search, elevation accuracy and the extent of the auxiliary calculations.

Argaman, Shamir and Spivak's program (1) had two decision variables - the fall across each stage and the drainage direction for each stage. The other researchers had one decision variable - the fall across each stage.

Researchers used different definitions for the state or elevation variable. The author considered the state to be the energy line elevation. Mays and Yen (26) used the crown elevation as the state. Meredith (27), Holland (19), Argaman and Spivak (1), Dajani and Hasit (7), Dajani, Gemmell and Morlok (6) used the invert elevation as the state. The author believes the energy line elevation is better because vertical alignment equations are normally based on the energy line.

Mays and Yen, and Meredith considered a manhole to be a stage consisting of a pipe of small length. The author considers this to be a disadvantage because it requires more data storage.

In Mays and Yen's DP and DDDP solution, for each feasible state, their program stored the state, diameter, length and slope of the pipe. In the author's program, only the best decision for each feasible state is stored. When the optimum decisions are retrieved, the pipes are redesigned and, in addition, complete vertical alignment data is computed.

None of the papers describe the logic of their manhole numbering system and order of solution. The author adopted a decimal numbering system for the manholes. This makes it easy to locate each manhole and provides a means of solution order and indexing data.

Dynamic programming is an effective tool for the solution of non-branching systems. Simple effective programs can be formulated for this problem, e.g. Meredith.

With a branching system, the solution is more complicated and much greater storage is required. Savings in storage could be effected if only the cheapest cost solution for each branch was stored. This appears to be the approach taken by Merritt and Bogan (28).

The author believes that his program is comparable with those of Merritt and Bogan, and Walsh and Brown (39) with the following exceptions:

1. Merritt and Bogan's program includes a pumping station,
2. Walsh and Brown have a unique method of accommodating critical elevations along the length of a sewer.

With regard to the pump option, it may be difficult to incorporate in the program appropriate pump head-discharge relationships and costs which would be necessary for the optimization. The author believes that pumping station requirements are usually identifiable, and it may be better to select the pumping station location and perform the pumping station calculations separately from the sewer optimization. Systems which require pumping stations can be divided into gravity sub-systems.

The author notes that the researchers quoted did not discuss the implications of cost vs diameter in cumulating branch costs.

The author used the cut state technique at a junction. This means that if there are NJUNCS junctions in the problem, there are NJUNCS + 1 serial systems in the problem. Mays and Yen (26) adopted a different approach because the cut state method "often has difficulty in defining the main chain and also requires large computer memory to store the computed information for the branches." In their method 'main' sewers are treated as a series of serial sewers. Thus, there would be 2*NJUNCS + 1 serial systems in the problem. The author agrees that this method would make for storage savings because the branch outlet solution data could be stored on the scratch file.

The author recognised that the DP approach would be expensive in execution time and so adopted a two pass system for his program. Mays and Yen used a DDDP approach to save execution time. They showed that there are considerable savings using their DDDP program as compared to their own DP program. For a 20-sewer, 4-junction problem, their DP program required 13.0S compilation time and 100.7s execution time; their DDDP program required for the most efficient solution shown, 17.1s compilation time and 13.1s execution time. The elevation accuracy of the DP solution was 0.02 ft. The DDDP solution was terminated by the minimum allowable cost difference; the elevation accuracy was of the order of 0.01 ft. Both of these elevation accuracies seem more than sufficient for a practical problem. For a 11-sewer, 3-junction problem the author's program required 17.5s complication time and 15.5s execution time[1]. However it is difficult to make an accurate comparison with other computer programs because computer efficiency is a function of the type of processor. In addition, one has to know the area of search and the extent of the auxiliary calculations. For example, the author's pipe cost equation was composed of exponential terms in the variables of cut and diameter, whereas Mays and Yen's pipe cost equation was composed of first order terms.

The author queries the value of the DDDP approach. The optimum state trajectory will normally lie close to the states defined by the minimum cover. A coarse grid search in the minimum cover area may be better. The time to convergence to an optimum solution depends on the initial trial trajectory which is selected by engineering judgement. If it is not clear where the optimum state trajectory may be located, the DDDP approach may require greater execution time than the author's two pass system.

CHAPTER 6.0 SENSITIVITY ANALYSIS

An analysis was performed to explore the sensitivity of the solution decision policy to the relative magnitude of the components in the pipe cost equation:

$$U = a + bC^d + eD^f$$

where,

U = cost per lineal foot ($)

C = depth to subgrade (ft)

D = pipe diameter (ft)

a = 9.73

b = .097

d = 1.6

e = .826

f = 2.387

In this type of equation, it is not possible to completely distinguish between the pipe cost vs excavation and backfill components. However, it was assumed that by varying coefficient e, an indication of the relative importance of pipe cost vs excavation and backfill costs would be obtained. All the other coefficients in the equation would remain unchanged.

An attempt was made to relate the effect of the e coefficient change to a dimensionless quantity, relative I, where I is defined by:

$$I = (\partial U/\partial D)/(\partial U/\partial C)$$

Let I for the derived cost equation be called I_d and I for other cost equations be called Io. The corresponding e values are e_d and e_o. Then relative I, Ir is given by:

$$I_r = I_o/I_d$$

$$= \frac{e_o}{e_d}$$

It was anticipated that if a range of relative I's could be identified in which there was no great change in the solution policy then a designer could prepare a preliminary sewer system design for tender, and select the final design using tendered unit prices with the assurance that the design policy would not change appreciably.

A 5-stage sanitary sewer serial system adapted from the Design and Construction of Concrete Sewers (11) was used for the sensitivity analysis. The sewer layout is shown in figure 6.1. The design criteria were:

1. level ground surface at elevation, 600.00,
2. density of population, 100 p.p.a.,
3. average rate of sewage flow, 200 i.g.p.c.d.,
4. Harmon formula for peak domestic flow,
5. infiltration rate, 1500 I.g.a.d.,
6. partial flow analysis,
7. minimum cover, 8 feet,

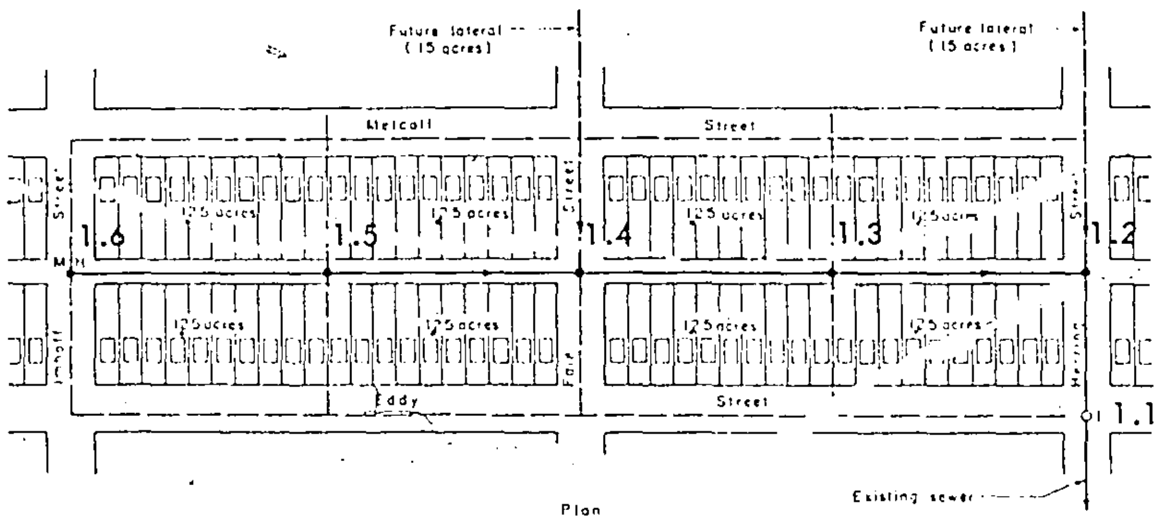


Figure 6.1 NETWORK FOR TEST PROBLEM 2

8. minimum full flow velocity, 2 f.p.s.
9. Manning's n = 0.013,
10. minimum pipe diameter, 8",
11. pipe diameters increased in 1" increments.

To reduce the effect of the discontinuity of commercial pipe diameters, the pipe diameters were increased in 1" increments.

The problem was solved with:

1. e = 0.826 (as per original data)
2. e = 2* 0.826
3. e = 0.5* 0.826

The optimum solution details are shown in table 6.1.

The pipe sizes selected for each solution reflects the interaction of the excavation and backfill vs pipe cost components.

The pipe supply and installation costs for a 15-inch diameter sewer in a cut of 12 feet are \$16.3, \$17.7 and \$15.6 for the cost equations used in solutions 1, 2 and 3 respectively. The pipe cost component, as given by the term containing e', is \$1.4, \$2.8 and \$0.7 respectively.

The results indicate that if the relative l value is 2, appreciable changes in the optimum outfall energy line elevation may be obtained. If the relative l value is 0.5, little change in the optimum outfall energy line elevation may be obtained; this is because excavation and backfill costs represent the major part of the system cost and decreasing the pipe cost component does not

Solution	Cost ($)	Relative Cost	Outfall ELELEV (ft.)	Pipes Used (in.)	Relative I
1	26,783	1	586.9	8 10 15 15 15	1
2	28,142	1.05	585.9	8 10 14 14 14	2
3	26,031	0.97	586.9	10 10 15 15 15	0.5

Table 6.1 SENSITIVITY ANALYSIS

materially effect the solution.

Further results are necessary to identify the range of relative I values in which there is minor change in the optimum outfall energy line elevation. In addition, the designer would have to satisfy himself that this range reflects the extent of variation to be expected in the tendered prices. In the event that this assurance was not possible, the designer may have to constrain the solution to meet mandatory elevations.

In conclusion, it may be stated that a more extensive analysis on a larger sewer system is necessary in order to obtain firm conclusions. In addition the cost equation should be composed only of the sum of the terms containing the two variables.

CHAPTER 7.0 SUMMARY AND RECOMMENDATIONS

Simultaneous optimization of sewer system layout, sizes and vertical alignment is, at present, not practical for large sewer systems.

Optimization of sewer sizes and vertical alignment for a fixed layout for large sewer systems is practical using the technique of dynamic programming. The optimization is particularly beneficial for use on sewer systems located in undulating terrain.

The uncertainty of obtaining reliable cost data detracts from the value of sewer optimization programs. However the author believes that such programs should be used because they present the most logical attempt to obtain the optimum sewer system cost. It may be possible to use tendered unit prices to finalize the system design.

Further work to improve the author's program should be directed to better utilisation of the arrays as discussed on page 119.

An analysis should be carried out to compare the efficiency of the author's two-pass method to the DDDP approach.

REFERENCES

1. Argaman Y., and Spivak E., "Design of Optimal Sewerage Systems", ASCE, Journal of Environmental Engineering Division, Vol. 99, No. EE5, Oct. 1973, pp. 703 - 716.

2. Argaman Y., Shamir U. and Spivak E., Closure of discussion on Ref. 1 , ASCE, Journal of Environmental Engineering, Vol. 101, No. EE2, April 1975, p. 266.

3. Baffa J., "Lateral Sewer Construction Costs", Public Works, Vol. 86, No. 11, pp. 71 - 77, 1955.

4. Barlow J. F., "Cost Optimization of Pipe Sewerage Systems", The Institution of Civil Engineers Proceedings, Part 2, Vol. 53, June 1972 pp. 57 - 64.

5. Bellman R., "Dynamic Programming", p. 83, Princeton University Press, Princeton, 1957.

6. Dajani J. S., Gemmell R.S., and Morlok E.K., "Optimal Design of Urban Wastewater Networks", Journal of the Sanitary Engineering Division, ASCE Vol. 98, No. 5A6, Dec. 1972, pp. 853 - 867.

7. Dajani J. S., and Hasit Y., "Capital Cost Minimization of Drainage Networks", ASCE, Journal of Environmental Engineering Division, Vol. 100, No. EE2, April 1974, Paper 10448, pp. 325 - 337.

8. Davis, C.V., and Sorenson K.E., "Handbook of Applied Hydraulics", Third edition, pp. 40 - 48, 40 - 49, McGraw-Hill, New York, 1969.

9. DeCicco P.R., Soehngen H.F., and Takagi J., "Use of Computers in Design of Sanitary Sewer Systems", Journal of the Water Pollution Control Federation, Vol. 40, No. 2. Pt 1, Feb. 1968, pp. 269 - 284.

10. Deininger R. A. "Computer Aided Design of Waste Collection and Treatment Systems", Proceedings of the Second Annual American Water Resources Conference, Nov. 1966, pp. 247 - 258.

11. "Design and Construction of Concrete Sewers", Portland Cement Association, Skokie, Illinois, 1968.

12. "Design and Construction of Sanitary and Storm Sewers", Manual on Engineering Practice, No. 37, Joint Committee of ASCE and the Water Pollution Control Federation, Washington, D.C., 1969.

13. Englesman C., "CE Cost Guide", 5th edition, CE Enterprises, Inc., Atlantic Heights, N.J., 1970.

14. Fair G.M., Geyer J.C., and Okun D.A., "Water Supply and Wastewater Removal", John Wiley & Sons, New York, 1966.

15. Fisher J.M., Karadi G.M., and McVinnie W.W., "Design of Sewer Systems", Water Resources Bulletin, Vol. 7, No. 2, Paper 71027 April 1971, pp. 294 - 302.

16. Haith D., "Vertical Alignment of Sewers and Drainage Systems by Dynamic Programming", thesis presented to the Massachusetts Institute of Technology at Cambridge, Mass., in 1966, in partial fulfillment of the requirements for the degree of Master of Science.

17. Heidari M., Chow V.T., Kokotović P.V., and Meredith D.D., "Discrete Differential Dynamic Programming Approach to Water Resources Systems Optimization", Water Resources Research, Vol. 7, No. 2, April 1971, pp. 273 - 282.

18. Holland M.E., Discussion of Ref.24, Journal of the Sanitary Engineering Division, Vol. 94, No. SA2, April 1968, p. 437.

19. Holland M.E., "Computer Models of Wastewater Collection Systems", Harvard Water Resources Group, Harvard University, Cambridge, Mass., 1966.

20. IBM System/360 and System/370 Fortran IV Language Reference Manual, Sample Program 2.

21. Khanna P., Discussion of Ref. 1 , ASCE, Journal of Environmental Engineering, Vol. 100, No. EE4, Aug. 1974, pp. 1042 - 1043.

22. Kühner J., and Harrington J.J., Discussion on Ref. 7, ASCE, Journal of Environmental Engineering, Vol. 101, No. EE2, April 1975, pp. 270 - 271.

23. Kuester J.L., and Mize J.H., "Optimization Techniques with Fortran", McGraw-Hill, New York, 1973.

24. Liebman J.C., "A Heuristic Aid for the Design of Sewer Networks", Journal of the Sanitary Engineering Division, ASCE, Vol. 93, No. SA4, Proc. Paper 5381, Aug. 1967, pp. 81 - 90.

25. Manual of Commercial and Industrial Construction Estimating and Engineering Standards, Vol. 1, Richardson Engineering Services, Inc., Downey California, 1971.

26. Mays L.W., and Yen B.C., "Optimal Cost of Branched Sewer Systems", Water Resources Research, Vol. II, No. 1, Feb. 1975, pp. 37 - 47.

27. Meredith D.D., Dynamic Programming With Case Study on Planning and Design of Urban Water Facilities, Treatise on Urban Water Systems, Sect. 9, pp. 590 - 652, Colo. State Univ., Fort Collins, July 1971.

28. Merritt L.B., and Bogan R.H., "Computer-Based Optimal Design of Sewer Systems", ASCE, Vol. 99, No. EE1, Feb. 1973, pp. 35 - 53.

29. "Minimum Velocities for Sewers", Final Rept. Comm. to Study Limiting Velocities of Flow in Sewers, Jour. Boston Soc. Civil Engr., 29, 286 (1942).

30. Mortenson R.R., "Large Water Transmission Main Installed Without the Need for Shoring", Public Works, Vol. 105, No. 12, Dec. 1974, p. 69.

31. Nemhauser G.L., "Introduction to Dynamic Programming", John Wiley, New York, 1966.

32. Pomeroy R.D., "Flow Velocities in Small Sewers", Journal of Water Pollution Control Federation, Vol. 39, No. 9, Sept. 1967, pp. 1525 - 1548.

33. Prim R.C., "Shortest Connexion Networks and some Generalizations", Bell Syst. Tech. J., 1957, 36, 1389 - 1401.

34. Rich L.G., "Environmental Systems Engineering", pp. 309 - 310, McGraw-Hill, New York, 1973.

35. Smith A.A., "Civil Engineering Program Library", McMaster University, 1975.

36. Tang W.H., Mays L.W., and Yen B.C., "Optimal Risk-Based Design of Storm Sewer Network", paper presented at 45th National ORSA/TIMS Meeting, April 1974.

37. Templeman A.B., and Wilson A.J., Discussion of Ref. 4, The Institution of Civil Engineers, Proceedings, Part 2, Vol. 53, Dec. 1972, pp. 639 - 646.

38. UWHAUS, MacGregor J.F., Dept. of Chemical Engineering, McMaster University.

39. Walsh S., and Brown L.C., "Least Cost Method of Sewer Design", ASCE, Journal of Environmental Engineering Division, Vol. 99, No. EE3, June 1973, pp. 333 - 345.

40. "Yardsticks for Costing 1974", Canadian Architect Supplement, July 1974.

41. Yen B.C., Tang W.H., and Mays L.W., "Design Storm Sewers Using the Rational Method", Part I - Water and Sewage Works, Vol. 121. No. 10, Oct. 1974, pp. 92 - 95, Part II - Vol. 121, No. 11, Nov. 1974, pp. 84 - 85.

APPENDIX 1

Program Listing

Name		Page
Program	SEWER	140
Function	PRICEP	143
Function	PRICED	143
Function	PRICES	143
Function	LOCATE	143
Subroutine	DESIGN	144
Subroutine	FLOW	167
Function	PIPE1	169
Function	PIPE2	171
Function	CSTPM	174
Function	COSTM	176

```
      PROGRAM SEWER(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE4)
C
C USER SUPPLIED PROGRAM
C PROGRAM TO READ IN USER, DATE, NUMBER OF PROBLEMS
C COST ARRAYS, SET ARRAY SIZES AND CALL DESIGN.
C LINES = MAXIMUM NUMBER OF SEWERS + 1
C NJMAX = MAXIMUM NUMBER OF JUNCTIONS
C NSMAX = MAXIMUM NUMBER OF STATES
C       = IFIX(FALL/0.1) + 1
C PROGRAM SIZE FOR FIELD LENGTH OF 116,000 WORDS =
C 99 SEWERS * 5 JUNCTIONS * 47.4 FT FALL
C OR VARIATIONS OF THE ABOVE PROVIDING NUMBER OF SEWERS
C IS LESS THAN 100
C NCUT = NUMBER OF CUTS CONSIDERED IN PIPE COST ARRAY
C NSHAFT = NUMBER OF VALUES IN SHAFT COST ARRAY
C NPIPES = MAXIMUM NUMBER OF PIPE SIZES CONSIDERED
C NBRNCH = MAXIMUM NUMBER OF BRANCHES
C
      COMMON CPLF(13,15),CBOX,DCLF(20),SCLF(25),ICOST,NI,NO,NTP,
     1 IUSER(5),IDAY,IMONTH,IYEAR,NPIPES,DIA(20)
C
C USER SUPPLIED DIMENSION STATEMENT
C
      DIMENSION COSTJ(5,200,2),
     1 DGELEV(15),DTEMP(200),
     2 ELCU(15),ELCL(15),ELELEV(15),
     3 FSBEST(200),FSOPT(200),
     4 ISTAGE(15),IELCU(15),IELCL(15),FOBEST(200),
     5 JNS(5),JU(15),JL(15),
     6 KTIES(15),KDEC(15),KBRNCH(6),
     7 PDIA(15),PENERGY(15),PCOST(15),PSLOPE(15),
     8 PQFULL(15),PVFULL(15),
     9 PVMIN(15),PVMAX(15),DEPTH(15),PDROP(15)
      DIMENSION PVHLSS(15),PINVDP(15),PFALL(15),
     1 PUINV(15),PDINV(15),PIPEJ(5,200,2),
     2 SDIAM(200),SDIAMJ(5,200),
     3 UGELEV(15),XJN(5)
C
C SET CONSTANTS
C
      NI=5
      NO=6
      NTP=4
      NCUT=15
      NSHAFT=25
      NPIPES=13
C
C SET USER SUPPLIED VARIABLES TO FIX ARRAY SIZES IN DESIGN
C NJMAX = 1, MINIMUM
C
      LINES=15
      NJMAX=5
      NBRNCH=NJMAX+1
      NSMAX=200
      LAYMAX=2
C
C INPUT THE DAY OF THE MONTH E.G. '21'
C INPUT THE MONTH E.G. 'AUG'
C INPUT THE YEAR E.G. '1974'
```

```
C INPUT USERS NAME
C
      READ (NI,10) IDAY,IMONTH,IYEAR,(IUSER(I),I=1,5)
10    FORMAT (I2,A3,I4,5A4)
C
C INPUT NUMBER OF PROBLEMS, NPROB, AND ICOST
C ICOST = 0, USE COST ARRAY DATA
C ICOST = 1, USE COST FUNCTION DATA
C
      READ (NI,20) NPROB,ICOST
20    FORMAT (2I2)
C
      IF (ICOST.EQ.1) GO TO 120
C
C COST ARRAY DATA, INPUT AND ECHO CHECK
C INPUT PIPE COST ARRAY
C INPUT SHAFT COST ARRAY
C INPUT DROP PIPE COST ARRAY
C INPUT MANHOLE BOX COST
C
      WRITE (NO,30) IDAY,IMONTH,IYEAR
30    FORMAT (1H1,4X,'COST ARRAY DATA - DATE',I3,1X,A3,1X,I5,/)
      WRITE (NO,40)
40    FORMAT (5X,'PIPE COST ARRAY',/,36X,'DEPTH (FT)',/,9X,'     6','
     17','     8','     9','    10','    11','    12','    14','    16','    18'
     2,'    20','    22','    24','    26','    28',/,5X,'DIA')
      DO 70 IROW=1,NPIPES
      READ (NI,50) IDIA,(CPLF(IROW,ICOL),ICOL=1,NCUT)
50    FORMAT (I4,15F5.1)
      DIA(IROW)=FLOAT(IDIA)
      WRITE (NO,60) IDIA,(CPLF(IROW,ICOL),ICOL=1,NCUT)
60    FORMAT (4X,I4,15F5.1)
70    CONTINUE
      READ (NI,80) (SCLF(I) I=1,NSHAFT)
80    FORMAT (8F10.5)
      WRITE (NO,90)
90    FORMAT (/,5X,'SHAFT COST ARRAY',/)
      WRITE (NO,100) (SCLF(I),I=1,NSHAFT)
100   FORMAT (5X,12F6.1)
      READ (NI,80) (DCLF(I) I=1,NPIPES)
      WRITE (NO,110)
110   FORMAT (/,5X,'DROP PIPE COST ARRAY',/)
      WRITE (NO,100) (DCLF(I),I=1,NPIPES)
      GO TO 130
C
120   CONTINUE
C
      READ (NI,80) (DIA(I),I=1,NPIPES)
130   CONTINUE
      READ (NI,80) CBOX
      WRITE (NO,160) CBOX
      WRITE (NO,140)
140   FORMAT (/,5X,'PIPE DIAMETERS (INCHES) CONSIDERED =')
      WRITE (NO,150) (DIA(I),I=1,NPIPES)
150   FORMAT (5X,12F6.0)
160   FORMAT (/,5X,'MANHOLE BOX COST =',F6.1)
      NRUN=1
170   CONTINUE
```

```
      CALL DESIGN(COSTJ,DGELEV,DIEMP,ELCU,ELCL,ELELEV,ESREST,
     1 ESCPT,ISTAGE,IELCU,IELCL,IPREST,JNS,JU,JL,KTIES,KDEC,
     2 KBRNCH,PDIA,PENERGY,PCOST,PSLOPE,PQFULL,PVFULL,PVMIN,
     3 PVMAX,PDEPTH,PPROB,PMHLSS,PINVOP,PFALL,PUINV,PDINV,PIPEJ,
     4 SDIAM,SDIAMJ,UGELEV,XJN,LINES,NJMAX,NBRNCH,NSMAX,LAYMAX,NRUN)
      NRUN=NRUN+1
      IF (NRUN.LE.NPROB) GO TO 170
      REWIND MTP
      STOP
      END
```

```
      FUNCTION PRICEP(CUT,DIA,PLNGTH)
C
C COMPUTES PRICE OF PIPE
C USER SUPPLIED FUNCTION
C CUT = DEPTH TO SUBGRADE (FT)
C DIA = PIPE DIAMETER (FT)
C PLNGTH = PIPE LENGTH (FT)
C
      UNITP = 9.73 + .007*CUT**1.6 + .926*DIA**2.987
      PRICEP=UNITP*PLNGTH
      RETURN
      END
      FUNCTION PRICED(HEIGHT,DIA)
C
C COMPUTES PRICE OF DROP PIPE
C USER SUPPLIED FUNCTION
C DIA = DIAMETER OF DROP PIPE (FT)
C HEIGHT = DROP HEIGHT (FT)
C
      UNITP=10.+10.*DIA
      PRICED=UNITP*HEIGHT
      RETURN
      END
      FUNCTION PRICES(HEIGHT)
C
C COMPUTES PRICE OF SHAFT
C USER SUPPLIED FUNCTION
C HEIGHT = SHAFT HEIGHT (FT)
C
      PRICES=121.18+37.91*HEIGHT+.088*HEIGHT**2-.009*HEIGHT**3
      RETURN
      END
      FUNCTION LOCATE(IJ,LOWEST,INCRMT)
C
C RETURNS THE IDENTIFICATION NUMBER FOR THE STATE
C
C
      LOCATE=(IJ-LOWEST+INCRMT)/INCRMT
      RETURN
      END
```

```
      SUBROUTINE DESIGN(COSTJ,DGELEV,DTEMP,ELCU,ELCL,ELELEV,FSBEST,
     1 FSOPT,ISTAGE,IELCU,IELCL,IDBEST,JNS,JU,JL,KTIES,KDEC,
     2 KBRNCH,PDIA,PENERGY,PCOST,PSLOPE,PQFULL,PVFULL,PVMIN,
     3 PVMAX,PDEPTH,PDROP,PMHLSS,PINVDP,PFALL,PUINV,PDINV,PIPEJ,
     4 SDIAM,SDIAMJ,UGELEV,XJN,LINES,NJMAX,NBRNCH,NSMAX,LAYMAX,NRUN)
C
C IN THE FIRST PASS THE PROGRAM USES 1 FT STATE INCREMENTS
C IN THE SECOND PASS THE PROGRAM USES 0.1 FT STATE INCREMENTS
C TYPE OF PIPE = ASTM C76 B WALL
C ALL ELEVATIONS AND MINIMUM COVER REQUIREMENTS TO
C FIRST DECIMAL PLACE
C NPIPES = MAXIMUM NUMBER OF PIPE SIZES CONSIDERED
C DIA(NPIPES) = DEFAULT PIPE SIZE
C             = 42 INCHES
C DIA(NPIPES - 1) = MAXIMUM OUTFALL DIAMETER
C                 = 39 INCHES
C DIA(1) = SMALLEST PIPE SIZE
C        = 8 INCHES
C
      COMMON CPLE(12,15),CROX,DCLF(20),SCLF(25),ICOST,NI,NO,NIP,
     1 IUSER(5),IDAY,IMONTH,IYEAR,NPIPES,DIA(20),
     2 ACRES(100),DMHN(100),DIAMAX,DEPTH,DROP,
     3 ENERGY,
     4 FALL,
     5 G,
     6 IUGS(100),IDGS(100),
     7 NLINES,NSTAGE,N,N1,NROW,
     8 PEAKES,POPD(100),PIPE,PI
      COMMON QADD(100),QDMAX(100),QDMIN(100),
     1 QFULL,
     2 RCOM,RDOM,RIND,RINF,RMANN,RUN(100),
     3 SLOPE,SIGN3,
     4 TYPE(100),
     5 UMHN(100),
     6 VMAX,VMIN,VFULL,VMAXP,VMINP
      DIMENSION COSTJ(NJMAX,NSMAX,LAYMAX),
     1 DGELEV(LINES),DTEMP(NSMAX),
     2 ELCU(LINES),ELCL(LINES),ELELEV(LINES),
     3 FSBEST(NSMAX),FSOPT(NSMAX),
     4 ISTAGE(LINES),IELCU(LINES),IELCL(LINES),IDBEST(NSMAX),
     5 IREM(12),JNS(NJMAX),JU(LINES),JL(LINES),
     6 KTIES(LINES),KDEC(LINES),KBRNCH(NBRNCH),
     7 PDIA(LINES),PENERGY(LINES),PCOST(LINES),PSLOPE(LINES),
     8 PQFULL(LINES),PVFULL(LINES),
     9 PVMIN(LINES),PVMAX(LINES),PDEPTH(LINES),PDROP(LINES)
      DIMENSION PMHLSS(LINES),PINVDP(LINES),PFALL(LINES),
     1 PUINV(LINES),PDINV(LINES),PIPEJ(NJMAX,NSMAX,LAYMAX),
     2 SDIAM(NSMAX),SDIAMJ(NJMAX,NSMAX),
     3 UGELEV(LINES),XJN(NJMAX)
      INTEGER HS,HD
      DATA IBLANK,ITIED1,ITIED2/3H   ,1H ,1H*/
C
C SET CONSTANTS
C
      LOWEST=0
      PI=3.1416
      G=32.174
      IBF=25
      IBF1=IBF-1
```

```
      BIGM=9.0E40
C
C IOPT3 = 0 , NO INTERMEDIATE RESULTS PRINTED
C IOPT3 = 1 ,INTERMEDIATE RESULTS PRINTED
C
      IOPT3=1
C
C SIGN3 = 0., FOR FULL FLOW ANALYSIS
C SIGN3 = 3., FOR PARTIAL FLOW ANALYSIS
C SIGN4 = 0., FOR 1 FT STATE INCREMENTS
C SIGN4 = 4., FOR .1 FT STATE INCREMENTS
C
      SIGN3=0.
      SIGN4=0.
      WRITE (NO,10)
10    FORMAT (1H1,///5X,'SANITARY SEWER DESIGN BY DYNAMIC PROGRAMMING')
C
C OUTPUT RUN,DATE AND USER
C
      WRITE (NO,20) NRUN,IDAY,IMONTH,IYEAR,(IUSER(I),I=1,5)
20    FORMAT (5X,'RUN',I3,5X,'DATE',I3,1X,A3,1X,I5,5X,'USER',5A4///)
C
C INPUT PROBLEM IDENTIFICATION
C
      READ (NI,30) (IREM(I),I=1,12)
30    FORMAT (12A4)
      WRITE (NO,40) (IREM(I),I=1,12)
40    FORMAT (5X,12A4///)
C
C PIPE DESIGN CRITERIA
C
      DIAMIN=8.
      RMANN=0.013
      VMIN=2.
      VMAX=15.
      COVMIN=7.
      COVMAX=25.
C
C PEAKFS = SIGNAL TO SELECT THE PEAKING FACTOR
C FOR HARMON FORMULA, PEAKFS = 0.
C FOR PEAKING FACTOR,5/P1000**0.2, PEAKFS = 1.
C
      PEAKFS=0.
C
C ANY PIPE DESIGN CRITERIA CHANGES
C ANSWER 'YES' OR LEAVE A BLANK CARD
C
      READ (NI,50) IYES
50    FORMAT (A3)
      IF (IYES.EQ.IBLANK) GO TO 70
C
C INPUT MINIMUM DIAMETER EG . '10.'
C INPUT MANNINGS N
C INPUT MINIMUM AND MAXIMUM VELOCITIES (FPS)
C INPUT MINIMUM AND MAXIMUM COVER (FT)
C INPUT PEAKING FACTOR SIGNAL, PEAKFS
C INPUT SIGN3 VALUE
C
```

```
      READ (NI,60) DIAMIN,RMANN,VMIN,VMAX,COVMIN,COVMAX,PEAKFS,SIGN3
60    FORMAT (8F10.5)
70    CONTINUE
C
C ECHO CHECK THE DESIGN CRITERIA
C
      WRITE (NO,80)
80    FORMAT (5X,'DESIGN CRITERIA',//)
      WRITE (NO,90) DIAMIN
 0    FORMAT (5X,'MINIMUM DIAMETER (INCHES) =',F4.0)
      WRITE (NO,100) RMANN
100   FORMAT (5X,'MANNINGS N =',F5.3)
      WRITE (NO,110) VMIN,VMAX
110   FORMAT (5X,'MINIMUM VELOCITY (FPS) ='F4.0/(5X,'MAXIMUM VELOCITY (F
     1PS) =',F4.0))
      WRITE (NO,120) COVMIN,COVMAX
120   FORMAT (5X,'MINIMUM COVER (FT) =',F4.1,/(5X,'MAXIMUM COVER (FT) ='
     1,F4.1))
      IF (PEAKFS-1.) 130,150,150
130   CONTINUE
      WRITE (NO,140)
140   FORMAT (5X,'PEAKING FACTOR = HARMON FORMULA')
      GO TO 170
150   CONTINUE
      WRITE (NO,160)
160   FORMAT (5X,'PEAKING FICTOR = 5./(POP**0.2)')
170   CONTINUE
      IF (SIGN3-3.) 180,200,180
180   CONTINUE
      WRITE (NO,190)
190   FORMAT (5X,'FULL FLOW CONDITIONS ONLY')
      GO TO 220
200   CONTINUE
      WRITE (NO,210)
210   FORMAT (5X,'NON FULL FLOW CONDITIONS CONSIDERED')
220   CONTINUE
C
C DESIGN FLOW CRITERIA
C
C DOMESTIC FLOW (IMP GALS PER CAPITA PER DAY)
C COMMERCIAL FLOW (IMP GALS PER ACRE PER DAY)
C INDUSTRIAL FLOW (IMP GALS PER ACRE PER DAY)
C INFILTRATION (IMP GALS PER ACRE PER DAY)
C
      RDOM=106.
      RCOM=3750.
      RIND=5000.
      RINF=2500.
C
C ANY CHANGES TO THE FLOW CRITERIA
C ANSWER 'YES' OR LEAVE A BLANK CARD
C
      READ (NI,50) IYES
      IF (IYES.EQ.IBLANK) GO TO 230
      READ (NI,60) RDOM,RCOM,RIND,RINF
230   CONTINUE
C
C ECHO CHECK THE DESIGN FLOWS
```

```
C
      WRITE (NO,240) RDOM,RCOM,RIND,RINF
240   FORMAT (5X,'DOMESTIC FLOW (IMP GALS/CAP/DAY) =',F4.0,/(5X,'COMMERC
     1IAL FLOW (IMP GALS/ACRE/DAY) =',F7.0),/(5X,'INDUSTRIAL FLOW (IMP G
     2ALS/ACRE/DAY) =',F7.0 ,/(5X,'INFILTRATION (IMP GALS/ACRE/DAY) =',F
     37.0)///)
C
C COLLECTION SYSTEM DATA
C
      WRITE (NO,250)
250   FORMAT (5X,'COLLECTION SYSTEM DATA',//)
C
C INPUT OUTFALL PIPE DIAMETER IN INCHES
C INPUT OUTFALL ENERGY LINE ELEVATION TO FIRST DECIMAL PLACE
C
      READ (NI,60) OUTDIA,OUTFAL
      WRITE (NO,260) OUTDIA,OUTFAL
260   FORMAT (5X,'OUTFALL DIAMETER (INCHES) =',F4.0/5X,'OUTFALL ENERGY L
     1INE ELEVATION =',F6.2,)
      DIAMAX=OUTDIA/12.+0.04
C
C INITIALIZE ISTAGE(I), UPPER ELEVATION CONSTRAINT ELCU(I)
C AND LOWER ELEVATION CONSTRAINT ELCL(I)
C ISTAGE(I) = 0, IF THERE ARE NO CONSTRAINT ELEVATIONS
C ISTAGE(I) = 1, IF THERE ARE CONSTRAINT ELEVATIONS
C I = SEWER NUMBER OR STAGE
C
      LM1=LINES-1
      DO 270 I=1,LM1
      ISTAGE(I)=0
      ELCU(I)=0.
      ELCL(I)=0.
270   CONTINUE
C
C FOR EACH SEWER,ENTER IN ORDER OF TYPICAL HAND CALCULATION
C UPSTREAM MANHOLE NUMBER, UMHN(I)
C DOWNSTREAM MANHOLE NUMBER, DMHN(I)
C UPSTREAM GROUND ELEVATION, UGELEV(I)
C DOWNSTREAM GROUND ELEVATION, DGELEV(I)
C LENGTH OF SEWER, RUN(I)
C ACREAGE TRIBUTARY TO THE SEWER, ACRES(I)
C POPULATION DENSITY, POPD(I)
C TYPE OF FLOW, TYPE(I)
C ADDITIONAL FLOW, QADD(I)
C UPPER AND LOWER CONSTRAINT ELEVATIONS, ELCU(I) AND ELCL(I)
C ALL ELEVATIONS TO THE FIRST DECIMAL PLACE
C LEAVE A BLANK CARD AT END OF DATA
C
      DO 290 I = 1,LINES
      READ(NI,280) UMHN(I),DMHN(I),UGELEV(I),DGELEV(I),RUN(I),ACRES(I),
     1 POPD(I),TYPE(I),QADD(I),ELCU(I),ELCL(I)
280   FORMAT (11F7.3)
      IF (UMHN(I).LT.1.) GO TO 300
290   CONTINUE
300   NLINES = I -1
C
C ECHO CHECK THE COLLECTION SYSTEM DATA
C
```

```
      WRITE (NO,310)
310   FORMAT (5X,'LINE',2X,'UMHN',4X,'DMHN',2X,'UGELEV',1X,'DGELEV',1X,'
     1RUN',2X,'ACRES',1X,'POPD',1X,'TYPE',1X,'QADD')
      WRITE (NO,320)
320   FORMAT (26X,'(FT)',3X '(FT)',2X,'(FT)',16X,'(CFS)',/)
      DO 330 I=1,NLINES
      WRITE (NO,340) (I,UMHN(I),DMHN(I),UGELEV(I),DGELEV(I),RUN(I),ACRES
     1(I),POPD(I),TYPE(I),QADD(I))
330   CONTINUE
340   FORMAT (5X,I3,2F8.3,2F7.2,F5.0,F5.1,F6.0,F4.0,F6.2)
      WRITE (NO,350)
350   FORMAT (5X,'DOMESTIC = 1.',4X,'COMMERCIAL = 2.',4X,'INDUSTRIAL = 3
     1.',///)
C
C ECHO CHECK THE CONSTRAINT ELEVATIONS
C
      NCONST=0
      DO 360 I=1,NLINES
      IF (ELCU(I).NE.0.) ISTAGE(I)=I
      IF (ELCL(I).NE.0.) NCONST=NCONST+1
360   CONTINUE
      IF (NCONST.LT.1) GO TO 410
      DO 400 I=1,NLINES
      IF (ISTAGE(I)-0) 370,390,370
370   CONTINUE
      WRITE (NO,380) ISTAGE(I),ELCU(I),ELCL(I)
380   FORMAT (5X,'STAGE =',I3,5X,'MINIMUM ENERGY LINE ELEVATION =',F8.2,
     1/20X,'MAXIMUM ENERGY LINE ELEVATION =',F8.2/)
390   CONTINUE
400   CONTINUE
410   CONTINUE
C
C
C  D E S I G N   S E C T I O N
C
C
C COMPUTE MAXIMUM AND MINIMUM FLOWS, QDMAX(I) AND QDMIN(I)
C
      CALL FLOW
      WRITE (NO,420) (I,QDMAX(I),I=1,NLINES)
420   FORMAT (5X,I3,4X,'QDMAX =',F5.2)
C
C FIND JUNCTION NUMBERS AND THE NUMBER OF PIPES IN THE BRANCHES
C KBRNCH(I) = NUMBER OF SEWERS IN BRANCH I
C XJN(I) = JUNCTION NUMBER
C
      DO 430 I=1,NBRNCH
      KBRNCH(I)=0
430   CONTINUE
      DO 440 NJUNC=1,NJMAX
      XJN(NJUNC)=0.
440   CONTINUE
      NJUNC=1
      NJUNCS=0
      LARGE=0
      DO 510 I=1,NLINES
      IUMHN=UMHN(I)
      IDMHN=DMHN(I)
```

```
      IF (IUMHN.GT.LARGE) LARGE=IUMHN
      KBRNCH(IUMHN)=KBRNCH(IUMHN)+1
      IF (IUMHN.EQ.IDMHN) GO TO 510
      IF (NJUNC-1) 450,490,450
450   CONTINUE
      DO 480 K=1,NJMAX
      IF (XJN(K).EQ.DMHN(I)) GO TO 500
      IF (XJN(K)-0.) 470,460,470
460   CONTINUE
      XJN(NJUNC)=DMHN(I)
      NJUNC=NJUNC+1
      GO TO 500
470   CONTINUE
480   CONTINUE
490   CONTINUE
      XJN(1)=DMHN(I)
      NJUNC=2
500   CONTINUE
510   CONTINUE
      NJUNCS=NJUNC-1
      IF (NJUNCS.EQ.0) NJUNCS=1
      DO 520 I=1,NLINES
      IUMHN=UMHN(I)
      IDMHN=DMHN(I)
      IF (IUMHN.NE.IDMHN) KBRNCH(IDMHN)=KBRNCH(IDMHN)+KBRNCH(IUMHN)
520   CONTINUE
C
C KSER = NUMBER OF SERIAL SYSTEMS OR BRANCHES
C
      KSER=LARGE
      KSERP1=KSER+1
C
C FIND HIGHEST MOST UPSTREAM GROUND ELEVATION
C AND CORRESPONDING BRANCH NUMBER, LLHIGH
C
      UGHIGH=0.
      DO 580 L=1,KSER
      LL=KSERP1-L
      DO 550 M=1,NLINES
      IUMHN=UMHN(M)
      IF (IUMHN.NE.LL) GO TO 540
      IF (UGELEV(M)-UGHIGH) 560,530,530
530   CONTINUE
      UGHIGH=UGELEV(M)
      LLHIGH=IUMHN
      GO TO 570
540   CONTINUE
550   CONTINUE
560   CONTINUE
570   CONTINUE
580   CONTINUE
C
C STATES ARE (DISTANCES FROM THE DATUM)*100
C MEASURE THE STATES POSITIVE DOWNWARDS
C TAKE THE STATE DATUM THROUGH THE ENERGY LINE ELEVATION
C AT THE HIGHEST MOST UPSTREAM GROUND ELEVATION
C FOR FULL FLOW CONDITIONS , THE INSIDE TOP OF THE PIPE
C FOR NON FULL FLOW CONDITIONS , THE CENTRE OF THE PIPE
```

```
C
      PWTMIN=(DIAMIN/12.+1.)/12.
      X=UGHIGH-COVMIN-PWTMIN
      IF (SIGN3.EQ.0.) X=UGHIGH-COVMIN-PWTMIN-0.5*DIAMIN/12.
C
C COMPUTATION STARTS HERE ON THE SECOND PASS
C
590   CONTINUE
      REWIND NTP
      MFLAG=0
C
C CONVERT THE GROUND ELEVATIONS TO STATES
C
      DO 630 I=1,NLINES
      UGS=-(UGELEV(I)-X)
      DGS=-(DGELEV(I)-X)
      IF (SIGN4-4.) 600,610,600
600   CONTINUE
      IF (UGS.LT.0.) UGS=UGS-0.5
      IF (DGS.LT.0.) DGS=DGS-0.5
      IF (UGS.GE.0.) UGS=UGS+0.5
      IF (DGS.GE.0.) DGS=DGS+0.5
      IUS=UGS
      IDS=DGS
      IUGS(I)=IUS*100
      IDGS(I)=IDS*100
      GO TO 620
610   CONTINUE
      UGS=UGS*10.
      DGS=DGS*10.
      IUS=UGS
      IDS=DGS
      IUGS(I)=IUS*10
      IDGS(I)=IDS*10
620   CONTINUE
630   CONTINUE
C
C COMPUTE THE MAXIMUM STATE  MAXS
C SET THE STATE AND DECISION INCREMENT, INCRMT
C SET THE LOW DECISION, LOWD
C
      OUT=-(OUTFAL-X)
      IF (SIGN4-0.) 650,640,650
640   CONTINUE
      OUT=OUT+0.5
      IOUT=OUT
      MAXS=IOUT*100
      INCRMT=100
      LOWD=100
      GO TO 660
650   CONTINUE
      OUT=OUT*10.
      IOUT=OUT
      MAXS=IOUT*10
      INCRMT=10
      LOWD=10
660   CONTINUE
C
```

```
C IF NUMBER OF STATES EXCEEDS THE ALLOWABLE, NSMAX
C TERMINATE THE JOB
C
      NNS=LOCATE(MAXS,LOWEST,INCRMT)
      IF (NNS-NSMAX) 690,690,670
670   CONTINUE
      WRITE (NO,680)
680   FORMAT (///5X,'JOB TERMINATED',/5X,'NUMBER OF STATES IS TOO LARGE'
     1)
      RETURN
690   CONTINUE
C
C INITIALIZE ARRAYS
C
      DO 700 I=1,NSMAX
      IDBEST(I)=8888
700   CONTINUE
      DO 730 NJUNC=1,NJUNCS
      DO 720 I=1,NNS
      SDIAMJ(NJUNC,I)=DIAMIN/12.
      DO 710 LAYER=1,LAYMAX
      COSTJ(NJUNC,I,LAYER)=0.
      PIPEJ(NJUNC,I,LAYER)=DIAMIN/12.
710   CONTINUE
720   CONTINUE
730   CONTINUE
C
      WRITE (NO,740)
740   FORMAT (15X,'STAGE',3X,'STATE',3X,'OPTIMUM VALUE',3X,'OPTIMUM DECI
     1SION',//)
C
C SELECT BRANCH LL FOR DESIGN
C
      DO 1610 L=1,KSER
      LL=KSERP1-L
C
C SET UPSTREAM STAGE N1 AND DOWNSTREAM STAGE NSTAGE
C FOR BRANCH LL
C
      DO 750 M=1,NLINES
      IUMHN=UMHN(M)
      IF (IUMHN.EQ.LL) N1=M
      IF (IUMHN.EQ.LL) GO TO 760
750   CONTINUE
760   CONTINUE
      NSTAGE=N1+KBRNCH(LL)-1
C
C DESIGN BRANCH LL
C
      DO 1600 N=N1,NSTAGE
      IUMHN=UMHN(N)
      IDMHN=DMHN(N)
      IF (IUMHN.NE.LL) GO TO 1430
C
C KOUNT1 = 0, IS SIGNAL TO CHECK IF OUTFALL SEWER HAS BEEN SELECTED
C KOUNT2 = 0, IS SIGNAL TO SELECT FIRST FEASIBLE PIPE
C DESIGNED IN ANY STAGE
C KOUNT3 = 3, IS SIGNAL TO SHOW THAT THERE IS A FEASIBLE
```

```
C PIPE FOR THE STAGE
C
      KOUNT1=0
      KOUNT2=0
      KOUNT3=0
C
C INITIALIZE FSBEST(I), IDBEST(I), DTEMP(I) AND SDIAM(I)
C
      DO 770 K=1,NNS
      FSBEST(K)=BIGM
      IDBEST(K)=8888
      IF (N.EQ.N1) DTEMP(K)=DIAMIN/12.
      IF (N.EQ.N1) SDIAM(K)=DIAMIN/12.
770   CONTINUE
C
C SET THE STATE VARIABLE NS = LOW STATE
C SET THE UPPER STATE VARIABLE LIMIT HS
C SET UPSTREAM STATE LIMITS JU AND JL FOR DOWNSTREAM STAGE
C
      IF (ISTAGE(N)-0) 810,780,810
C
780   CONTINUE
C
C FACT = APPROXIMATE DISTANCE BETWEEN THE ENERGY LINE
C ELEVATION AND THE TOP OF THE PIPE (OUTSIDE DIAMETER)
C COMPUTE FACT FOR STAGE N1
C FACT IS COMPUTED FROM UPSTREAM STAGE FOR OTHER STAGES
C
      IF (N-N1) 800,790,800
790   CONTINUE
      FACT=PWTMIN
      IF (SIGN3.EQ.3.) FACT=PWTMIN+0.5*DIAMIN/12.
800   CONTINUE
      XNS=-(DGELEV(N)-X)+COVMIN+FACT
      XHS=-(DGELEV(N)-X)+COVMAX
      XNS=XNS+0.5
      XHS=XHS+0.5
      NS=XNS
      HS=XHS
      NS=NS*100
      HS=HS*100
      IF (HS.GT.MAXS) HS=MAXS
      IF (LL.EQ.1.AND.N.EQ.NSTAGE) HS=MAXS
      JU(N+1)=NS
      JL(N+1)=HS
      GO TO 850
C
810   CONTINUE
C
      ELEVCU=-(ELCU(N)-X)
      ELEVCL=-(ELCL(N)-X)
      IF (SIGN4-0.) 830,820,830
820   CONTINUE
      ELEVCU=ELEVCU+0.5
      ELEVCL=ELEVCL+0.5
      IELU=ELEVCU
      IELL=ELEVCL
      IELCU(N)=IELU*100
```

```
      IELCL(N)=IELL*100
      GO TO 840
830   CONTINUE
      ELEVCU=ELEVCU*10.
      ELEVCL=ELEVCL*10.
      KCU=ELEVCU
      KCL=ELEVCL
      IELCU(N)=KCU*10
      IELCL(N)=KCL*10
840   CONTINUE
      NS=IELCU(N)
      HS=IELCL(N)
      IF (HS.GT.MAXS) HS=MAXS
      JU(N+1)=NS
      JL(N+1)=HS
850   CONTINUE
C
C SET UPSTREAM STATE LIMITS FOR STAGE N1
C DO THIS ON FIRST PASS ONLY
C
      IF (SIGN4-4.) 860,900,860
860   CONTINUE
      IF (N-N1) 910,870,910
870   CONTINUE
      IF (IUMHN-LLHIGH) 890,880,890
880   CONTINUE
      JU(N)=LOWEST
      JL(N)=HS-LOWD
      GO TO 920
890   CONTINUE
      FACT=PWTMIN
      IF (SIGN3.EQ.3.) FACT=PWTMIN+0.5*DIAMIN/12.
      XLOW=-(UGELEV(N)-X)+COVMIN+FACT
      XLOW=XLOW+0.5
      LOW=XLOW
      LOW=LOW*100
      JU(N)=LOW
      JL(N)=HS-LOWD
900   CONTINUE
910   CONTINUE
920   CONTINUE
C
C IF THE LOWEST STATE IS GREATER THAN THE OUTFALL STATE
C TERMINATE THE JOB
C
      IF (NS-MAXS) 940,940,930
930   CONTINUE
      WRITE (NO,970) DMHN(N)
      RETURN
940   CONTINUE
C
C SET THE DECISION VARIABLE ND = LOW DECISION
C SET THE UPPER DECISION VARIABLE LIMIT HD
C
      ND=LOWD
      IF (SIGN4-4.) 950,990,950
950   CONTINUE
      XMAXD=(COVMAX-COVMIN) (UGELEV(N)-DGELEV(N))
```

```
C
C IF THE AVAILABLE DECISION IS NEGATIVE, TERMINATE THE JOB
C
      IF (XMAXD) 960,960,980
960   CONTINUE
      WRITE (NO,970) DMMN(N)
970   FORMAT (///5X,'JOB TERMINATED AT MANHOLE NUMBER = ',F8.2/5X,'INSUF
     1FICIENT FALL')
      REWIND NTP
      RETURN
980   CONTINUE
C
      MAXD=IFIX(XMAXD*100.)
      IF (LL.EQ.1.AND.N.EQ.NSTAGE) MAXD=MAXS-IUGS(N)-IFIX(COVMIN*100.)
      HD=MAXD
      GO TO 1070
C
C IF THE PROGRAM IS ON THE SECOND PASS
C USE ELEVATION CONSTRAINTS TO SET HD
C
990   CONTINUE
      IF (N-N1) 1030,1000,1030
1000  CONTINUE
      IF (LL-LLHIGH) 1020,1010,1020
1010  CONTINUE
      HD=IELCL(N1)
      JU(N)=LOWEST
      JL(N)=HS-LOWD
      GO TO 1060
1020  CONTINUE
      PWT=(PDIA(N1)/12.+1.)/12.
      FACT=PWT
      IF (SIGN3.EQ.3.) FACT=PWT+PDIA(N1)/12.-PENERGY(N1)
      XLOW=-(UGELEV(N1)-X)+COVMIN+FACT
      XLOW=XLOW*10.
      LOW=XLOW
      LOW=LOW*10
      IF (LOW.LT.LOWEST) LOW=LOWEST
      HD=IELCL(N1)-LOW
      JU(N)=LOW
      JL(N)=HS-LOWD
      GO TO 1060
1030  CONTINUE
C
C CHECK THAT UPSTREAM SEWER IS NOT A DUMMY SEWER
C
      NSTMN1=NSTAGE-N1
      DO 1040 K=1,NSTMN1
      NMK=N-K
      INMK=UMHN(NMK)
      IF (INMK.EQ.LL) GO TO 1050
1040  CONTINUE
1050  CONTINUE
      HD=IELCL(N)-IELCU(NMK)
1060  CONTINUE
1070  CONTINUE
C
C SORT THE JUNCTION COSTS AND PIPE SIZES
```

```
C IF THE SEWER DOWNSTREAM OF THE JUNCTION IS BEING DESIGNED
C
      DO 1080 NJUNC=1,MJUNC
      IF (UMHN(N).EQ.XJN(NJUNC)) GO TO 1090
1080  CONTINUE
      GO TO 1140
1090  CONTINUE
      DO 1130 LAYER=1,LAYMAX
      IF (COSTJ(NJUNC,1,LAYER).EQ.0.) GO TO 1120
      SMALL=BIGM
      DO 1110 M=1,MNS
      IF (SMALL.EQ.BIGM.AND.COSTJ(NJUNC,M,LAYER).GE.BIGM) GO TO 1100
      IF (COSTJ(NJUNC,M,LAYER).LT.SMALL) SMALLP=PIPEJ(NJUNC,M,LAYER)
      IF (COSTJ(NJUNC,M,LAYER).LT.SMALL) SMALL=COSTJ(NJUNC,M,LAYER)
      IF (COSTJ(NJUNC,M,LAYER).GT.SMALL) PIPEJ(NJUNC,M,LAYER)=SMALLP
      IF (COSTJ(NJUNC,M,LAYER).GT.SMALL) COSTJ(NJUNC,M,LAYER)=SMALL
      IF (PIPEJ(NJUNC,M,LAYER).GT.SDIAMJ(NJUNC,M)) SDIAMJ(NJUNC,M)=PIPEJ
     1(NJUNC,M,LAYER)
1100  CONTINUE
1110  CONTINUE
1120  CONTINUE
1130  CONTINUE
1140  CONTINUE
1150  I=LOCATE(NS,LOWEST,INCRMT)
C
C COMPUTE THE PIPE SIZE
C
1160  CONTINUE
      NSMND=NS-ND
C
C IF NSMND IS NOT FEASIBLE, GO TO NEXT DECISION
C
      IF (NSMND.LT.JU(N).OR.NSMND.GT.JL(N)) GO TO 1250
C
C SET SDIA = UPSTREAM PIPE DIAMETER
C SET SDIA = JUNCTION PIPE DIAMETER, IF APPLICABLE
C
      II=LOCATE(NSMND,LOWEST,INCRMT)
      SDIA=DTEMP(II)
      SDIAJ=0.
      IF (UMHN(N).EQ.XJN(NJUNC)) SDIAJ=SDIAMJ(NJUNC,II)
      IF (SDIA.LT.SDIAJ) SDIA=SDIAJ
      IF (SIGN3.EQ.0.) PIPE=PIPE1(N,ND,SDIA)
      IF (SIGN3.EQ.3.) PIPE=PIPE2(N,ND,SDIA)
C
C COMPUTE FACT FOR USE IN SETTING LOW STATE IN DOWNSTREAM STAGE
C
      IF (KOUNT2-0) 1180,1170,1180
1170  CONTINUE
      KOUNT2=2
      PWT=(PIPE+1.)/12.
      FACT=PWT
      IF (SIGN3.EQ.3.) FACT=PWT+PIPE-ENERGY
C
C IF THE DEFAULT PIPE WAS CHOSEN, SET KOUNT2 = 0
C
      IF (ABS(PIPE-(DIA(NPIPES)/12.)).LT.0.04) KOUNT2=0
1180  CONTINUE
```

```
C
C COMPUTE THE COST FOR THIS STAGE
C
      COST=CSTFN(N,NS,ND)
C
C
C CHECK IF DESIGN IS ON STAGE N1
C
      IF (N-N1) 1200,1190,1200
1190  CONTINUE
      FSTAR=0.
      GO TO 1210
1200  CONTINUE
      FSTAR=FSOPT(II)
1210  CONTINUE
C
C CUMULATE COSTS
C
      FNSXN=COST+FSTAR
C
C ADD THE TRIBUTARY BRANCH COSTS IF APPLICABLE
C
      IF (XJN(NJUNC).EQ.UMHN(N)) FNSXN=FNSXN+COSTJ(NJUNC,II,1)+COSTJ(NJU
     1NC,II,2)
C
C FLAG A TIED SOLUTION
C
      ITIED=0
      IF (FNSXN.EQ.FSBEST(I)) ITIED=1
C
C DISCARD SOLUTIONS WITH DIAMETER GREATER THAN OUTFALL DIAMETER
C
      IF (PIPE-DIAMAX) 1220,1220,1260
1220  CONTINUE
      KOUNT3=3
C
C SAVE THE CHEAPEST COST ,THE DECISION AND THE PIPE SIZE
C TIES BROKEN IN FAVOUR OF LOWEST VALUED DECISION
C
      IF (FNSXN-FSBEST(I)) 1230,1240,1240
1230  FSBEST(I)=FNSXN
      SDIAM(I)=PIPE
      IDBEST(I)=ND
C
C FOR TIED SOLUTIONS, MAKE IDBEST(I) = - IDBEST(I)
C
1240  CONTINUE
      IF (ITIED.EQ.0) GO TO 1270
      IX=IDBEST(I)
      IDBEST(I)=-IABS(IX)
C
C INCREMENT THE DECISION VARIABLE ND
C
1250  CONTINUE
1260  CONTINUE
1270  ND=ND+INCRMT
      IF (ND.LE.HD) GO TO 1160
      IF (IOPT3.LE.0) GO TO 1310
```

```
C
C OUTPUT STAGE, STATE, COST, AND DECISION
C OPTIONAL EXCEPT FOR THE OUTFALL STAGE
C
1280  CONTINUE
      IX=IDBEST(I)
      ITIED=ITIED1
      IF (IX.GE.0) GO TO 1290
      IX=-IX
      ITIED=ITIED2
      MFLAG=1
1290  WRITE (NO,1300) M,NS,FOBEST(I),IX,ITIED
1300  FORMAT (16X,I2,5X,I4,5X,F12.2,8X,I4,A1)
      IF (KOUNT1.NE.0) GO TO 1600
C
C INCREMENT THE STATE VARIABLE NS
C RE-ASSIGN ND
C
1310  CONTINUE
      NS=NS+INCRMT
      ND=LOWD
      IF (NS-HS) 1150,1150,1320
1320  CONTINUE
C
C IF THERE WAS NO SUITABLE SEWER FOR THIS STAGE
C TERMINATE THE JOB
C
      IF (KOUNT2-3) 1330,1350,1350
1330  CONTINUE
      WRITE (NO,1340) DMHN(N)
1340  FORMAT (///5X,'JOB TERMINATED AT MANHOLE NUMBER =',F8.2/5X,'OUTFAL
     1L DIAMETER IS TOO SMALL')
      REWIND NTP
      RETURN
1350  CONTINUE
C
C IF DESIGN IS ON THE FINAL STAGE OF THE MAIN BRANCH, GO TO
C ROUTINE WHICH SELECTS THE CHEAPEST SEWER SYSTEM COST
C
      IF (M.EQ.NSTAGE.AND.LL.EQ.1) GO TO 1510
C
      IF (IUMHN-IDMHN) 1360,1420,1360
C
1360  CONTINUE
C
C STORE THE PIPE SIZE AT THE JUNCTION
C STORE THE COSTS AT THE JUNCTION
C
      DO 1370 NJUNC=1,NJUNC
      IF (XJN(NJUNC).EQ.DMHN(M)) GO TO 1380
1370  CONTINUE
1380  CONTINUE
      LAYER=-1
      DO 1390 M=1,NLINES
      IF (XJN(NJUNC).EQ.DMHN(M)) LAYER=LAYER+1
      IF (UMHN(M).EQ.UMHN(N)) GO TO 1400
1390  CONTINUE
1400  CONTINUE
```

```
      DO 1410 M=1,NNS
      COSTJ(NJUNC,M,LAYER)=FSBEST(M)
      PIPEJ(NJUNC,M,LAYER)=SDIAM(M)
1410  CONTINUE
1420  CONTINUE
      GO TO 1450
C
1430  CONTINUE
C
C ASSIGN MAXIMUM UPSTREAM STATE LIMITS FOR DOWNSTREAM STAGE
C
      IF(JL(N).GT.JL(N+1)) JL(N+1) = JL(N)
      IF(JU(N).GT.JU(N+1)) JU(N+1) = JU(N)
C
C MAKE IDBEST(M) = 0 FOR THIS STAGE
C
      DO 1440 M=1,NNS
      IDBEST(M)=0
1440  CONTINUE
C
1450  CONTINUE
C
C SAVE THE OPTIMUM DECISION  FOR THIS STAGE ON TAPE
C
      DO 1470 KTAPE=1,NNS
      J=(KTAPE-1)*IBF+1
      J34=J+IBF1
      WRITE (NTP,1460) (IDBEST(M),M=J,J34)
1460  FORMAT (1X,25I5)
      IF (J34.GE.NNS) GO TO 1480
1470  CONTINUE
1480  CONTINUE
      IF (IDBEST(I).EQ.0) GO TO 1600
C
C PRINT PIPE SIZES FOR THIS STAGE
C
      WRITE (NO,1490) (SDIA (K),K=1,NNS)
1490  FORMAT (13F5.2)
C
      IF (N.EQ.NSTAGE) GO TO 1600
C
C UPDATE FSOPT(K) AND DTEMP(K) FOR USE AT THE NEXT STAGE
C
      DO 1500 K=1,NNS
      DTEMP(K)=SDIAM(K)
      FSOPT(K)=FSBEST(K)
1500  CONTINUE
      GO TO 1600
C
1510  CONTINUE
C
C PRINT THE PIPE SIZES FOR THIS STAGE
C
      WRITE (NO,1490) (SDIA (K),K=1,NNS)
C
C FIND THE CHEAPEST COST FOR THE SEWER SYSTEM
C
      IMAX=1
```

```
      IF (ISTAGE(N)-0) 1530,1520,1530
1520  CONTINUE
      NS=XNS
      NS=NS*100
      GO TO 1540
1530  CONTINUE
      NS=IFLOW(N)
1540  CONTINUE
      IMIN=LOCATE(NS,LOWEST,INCRMT)
      FSBESL=BIGN
      IDBESL=9999
      DO 1570 J=IMIN,IMAX
      IF (FSBEST(J)-FSBESL) 1550,1560,1560
1550  CONTINUE
      FSBESL=FSBEST(J)
      IDBESL=IDBEST(J)
      KOUNT1=J
1560  CONTINUE
1570  CONTINUE
C
C IF NO COST WAS SELECTED, TERMINATE THE JOB
C
      IF (KOUNT1.NE.0) GO TO 1590
      WRITE (NO,1580)
1580  FORMAT (///5X,'JOB TERMINATED AT OUTFALL MANHOLE')
      REWIND NTP
      RETURN
1590  CONTINUE
C
      IDBEST(KOUNT1)=IDBESL
      FSBEST(KOUNT1)=FSBESL
      I=KOUNT1
      NS=(KOUNT1*INCRMT)+LOWEST-INCRMT
      GO TO 1280
1600  CONTINUE
1610  CONTINUE
C
      IF (MFLAG.GT.0) WRITE (NO,1620)
1620  FORMAT (/15X,'* AFTER THE OPTIMUM DECISION INDICATES',/15X,'THAT A
     1LTERNATE OPTIMAL DECIOIONS EXIST')
C
C
C  R E C O V E R Y   S E C T I O N
C
C
C SELECT BRANCH L FOR DESIGS
C
      DO 2250 L=1,KSER
      IF (L-1) 1630,1750,1630
1630  CONTINUE
C
C FIND THE STATE ELEVATION OF THE MAIN SEWER
C AT THE DOWNSTREAM END OF BRANCH L
C
      DO 1640 LL=1,NLINES
      IUMHN=UMHN(LL)
      IF (IUMHN.EQ.L) M1=LL
      IF (IUMHN.EQ.L) GO TO 1650
```

```
1640  CONTINUE
1650  CONTINUE
      NSTAGE=N1+KBRNCH(L)-1
      DO 1660 NJUNC=1,MJUNC
      IF (DMUN(NSTAGE).EQ.YJN(NJUNC)) NS=JNS(NJUNC)
      IF (DMUN(NSTAGE).EQ.YJN(NJUNC)) GO TO 1670
1660  CONTINUE
1670  CONTINUE
C
C FIND THE LOWEST STATE WHICH GIVES THE CHEAPEST COST
C
      LAYER=-1
      DO 1680 M=1,MLINES
      IF (XJN(NJUNC).EQ.DMUN(M)) LAYER=LAYER+1
      IF (UMUN(M).EQ.UMUN(NSTAGE)) GO TO 1690
1680  CONTINUE
1690  CONTINUE
      I=LOCATE(NS,LOWEST,INCRMT)
      II=I
      SMALL=COSTJ(NJUNC,I,LAYER)
      DO 1700 M=1,MNS
      I=II-M
      IF (COSTJ(NJUNC,I,LAYER).LE.SMALL) GO TO 1700
      GO TO 1710
1700  CONTINUE
1710  CONTINUE
      I=I+1
      NS=(I*INCRMT)+LOWEST-INCRMT
C
C BACKSPACE TAPE TO THE FRONT OF STAGE, NSTAGE
C
      DO 1720 M=1,KTAPE
      BACKSPACE NTP
1720  CONTINUE
C
C READ THE OPTIMUM DECISIONS FOR STAGE, NSTAGE
C SELECT THE BEST DECISION
C
      DO 1730 M=1,KTAPE
      BACKSPACE NTP
1730  CONTINUE
      DO 1740 K=1,KTAPE
      KK=(K-1)*IBE+1
      K34=KK+IBE1
      READ (NTP,1460) (IDBEST(J),J=KK,K34)
1740  CONTINUE
      IX=IDBEST(I)
C
1750  CONTINUE
      N2=NSTAGE
      JS=NS
      M2=N1
      KDEC(M2)=IX
      NTY=0
      IF (IDBEST(I).GT.0) GO TO 1810
      NTY=NTY+1
      KTIES(NTY)=N2
      GO TO 1810
```

```
C
C BACKSPACE TAPE TO POSITION IN FRONT OF NEXT STAGE
C
1760  DO 1770 K=1,KTAPE
      BACKSPACE NTP
1770  CONTINUE
      DO 1780 K=1,KTAPE
      BACKSPACE NTP
1780  CONTINUE
C
C READ THE OPTIMUM DECISION  FOR THE PREVIOUS STAGE
C SELECT THE BEST DECISION
C
1790  DO 1800 K=1,KTAPE
      KK=(K-1)*IBE+1
      K24=KK+IBE1
      READ (NTP,1460) (IDBEST(J),J=KK,K24)
1800  CONTINUE
      KSS=LOCATE(JS,LOWEST,INCRMT)
      N2=N2-1
      M2=M2+1
      KDEC(M2)=IDBEST(KSS)
      IF (KDEC(M2).EQ.0) GO TO 1760
      IF (KDEC(M2).GT.0) GO TO 1810
      KDEC(M2)=-KDEC(M2)
      NTY=NTY+1
      KTIES(NTY)=N2
1810  CONTINUE
      IUMHN=UMHN(N2)
      IDMHN=DMHN(N2)
      IF (IUMHN.NE.IDMHN) GO TO 1830
C
C STORE THE STATE, JS, AT THIS JUNCTION
C IN THE ARRAY, JNS(NJUNC)
C
      DO 1820 NJUNC=1,NJUNC
      IF (DMHN(N2).EQ.XJN(NJUNC)) JNS(NJUNC)=JS
1820  CONTINUE
C
1830  CONTINUE
      IF (N2.EQ.N1) GO TO 1890
      JX=KDEC(M2)
C
      IF (JX-8888) 1860,1840,1860
1840  CONTINUE
      WRITE (NO,1850) L,M2
1850  FORMAT (5X,'ERROR ON BRANCH =',I5,'AT STAGE =',I5)
      REWIND NTP
      RETURN
1860  CONTINUE
C
      JS=JS-JX
      IF (L.NE.1.OR.N2.NE.NSTAGE) GO TO 1760
C
C PUT THE TAPE IN READING MODE
C
      REWIND NTP
      NSEWER=0
```

```
      DO 1870 I=1,KSER
      NSEWER=NSEWER+KBRNCH(I)
 1870 CONTINUE
      IF (NSEWER.LT.2) GO TO 1790
      KR=(NSEWER-2)*KTAPE
      DO 1880 K=1,KR
      READ (NTP,1460) (IPBEST(J),J=1,IPE)
 1880 CONTINUE
      GO TO 1790
C
C PRINT THE OPTIMAL DECISION FOR EACH STAGE
C
 1890 N3=NSTAGE+1
      WRITE (NO,1900) (KDEC(K),N3-K,K=N1,NSTAGE)
 1900 FORMAT (5(/14X,5(I4,'=X',I2,1X)))
      WRITE (NO,1910)
 1910 FORMAT (78X,1H.)
      IF (NTY.EQ.0) GO TO 1930
      WRITE (NO,1920) (KTIES(K),K=1,NTY)
 1920 FORMAT (15X,'ALTERNATE OPTIMAL DECISIONS AT STAGE/S',2I3,/15X,17I3
     1)
      WRITE (NO,1910)
 1930 CONTINUE
C
C RECALCULATE THE OPTIMUM PIPE DATA
C
      N=N1
      KC=1
      IUMHN=UMHN(N)
      IF(IUMHN.EQ.1) TCOST = 0.
      IF (N1.EQ.NSTAGE) GO TO 1980
 1940 CONTINUE
C
C FIND THE STATE NS FOR SEWER N
C
      IF (N-N1) 1950,1960,1950
 1950 CONTINUE
      NS=NS+KDEC(N3-KC)
      GO TO 1990
 1960 CONTINUE
      ISUM=0
      NSTM1=NSTAGE-1
      DO 1970 M=N1,NSTM1
      ISUM=ISUM+KDEC(M)
 1970 CONTINUE
      NS=NS-ISUM
 1980 CONTINUE
 1990 CONTINUE
C
C DESIGN SEWER N
C
      KDECK=KDEC(N3-KC)
C
C FIND SMALLEST DIAMETER ALLOWABLE, SDIA
C
      IF (N-N1) 2010,2000,2010
 2000 CONTINUE
      SDIA=DIAMIN/12.
```

```
      GO TO 2030
 2010 CONTINUE
      NSUP=NS-KDECK
      IUP=LOCATE(NSUP,LOWEST,INCRMT)
      SDIA=DTEMP(IUP)
      SDIAJ=0.
      DO 2020 NJUNC=1,NJUNC
      IF (UMHM(N).EQ.XJN(NJUNC)) SDIAJ=SDIANJ(NJUNC,IUP)
 2020 CONTINUE
      IF (SDIA.LT.SDIAJ) SDIA=SDIAJ
 2030 CONTINUE
      IF (SIGN3.EQ.0.) PIPE=PIPE1(N,KDECK,SDIA)
      IF (SIGN3.EQ.2.) PIPE=PIPE2(N,KDECK,SDIA)
      COST=CSTPM(N,NS,KDECK)
      TCOST=TCOST+COST
      PCOST(N)=COST
      KK=LOCATE(NS,LOWEST,INCRMT)
      DTEMP(KK)=PIPE
      PSLOPE(N)=SLOPE
      PDIA(N)=PIPE*12.
      PQFULL(N)=QFULL
      PVFULL(N)=VFULL
      PVMIN(N)=VMINP
      PVMAX(N)=VMAXP
      PDEPTH(N)=DEPTH
      PENERGY(N)=ENERGY
      PDROP(N)=DROP
      PFALL(N)=FALL
C
C COMPUTE THE ENERGY LOSS ELOSS
C
      IF (SIGN3.EQ.0.) GO TO 2070
      IF (N-N1) 2040,2050,2040
 2040 CONTINUE
      VU=PVMAX(N-1)
      VL=PVMAX(N)
C DAVIS P40-40
      IF (VL.GE.VU) ELOSS=0.2*(VL**2-VU**2)/(2.*G)
      IF (VL.LT.VU) ELOSS=0.2*(VU**2-VL**2)/(2.*G)
      IF (ELOSS.LT.0.02) ELOSS=0.02
      GO TO 2060
 2050 CONTINUE
      ELOSS=0.
 2060 CONTINUE
      IF (DROP.NE.0.) ELOSS=0.
      PMHLSS(N)=ELOSS
 2070 CONTINUE
      IF (SIGN3.EQ.0.) PMHLSS(N)=0.
C
C COMPUTE THE INVERT DROP PINVDP(N)
C
      IF (N-N1) 2080,2090,2080
 2080 CONTINUE
```

```
      PINVPP(N)=(PDIA(N)-PDIA(N-1))/12.+DROP
      IF (SIGN3.EQ.3.) PINVPP(N)=PENERGY(N)-PENERGY(N-1)+FLOSS+DROP
      IF (PINVPP(N).LT.0.)  INVPP(N)=0.
      GO TO 2100
 2090 CONTINUE
      PINVPP(N)=0.+DROP
 2100 CONTINUE
C
C COMPUTE THE INVERT ELEVATIONS PUINV(N) AND PDINV(N)
C
      IF (SIGN3-3.) 2110,2120,2110
 2110 CONTINUE
      PUINV(N)=FLOAT(NS-KDEC(N3-KC))/100.+PIPE+DROP
      PDINV(N)=FLOAT(NS)/100.+PIPE
      GO TO 2160
 2120 CONTINUE
      IF (N-N1) 2130,2140,2130
 2130 CONTINUE
      PUINV(N)=FLOAT(NS-KDEC(N3-KC))/100.+ENERGY+FLOSS+DROP
      PDINV(N)=FLOAT(NS)/100.+ENERGY
      GO TO 2150
 2140 CONTINUE
      PUINV(N)=FLOAT(NS-KDEC(N3-KC))/100.+ENERGY+DROP
      PDINV(N)=FLOAT(NS)/100.+ENERGY
 2150 CONTINUE
 2160 CONTINUE
      PUINV(N)=X-PUINV(N)
      PDINV(N)=X-PDINV(N)
      IF (N.EQ.N1) GO TO 2170
      IF (PUINV(N).GT.PDINV(N-1)) PUINV(N)=PDINV(N-1)
 2170 CONTINUE
C
C COMPUTE THE ENERGY LINE ELEVATION ELELEV(N)
C
      ELELEV(N)=PDINV(N)+PIPE
      IF (SIGN3.EQ.3.) ELELEV(N)=PDINV(N)+ENERGY
      N=N+1
      KC=KC+1
      IF (N.GT.NSTAGE) GO TO 2240
 2180 CONTINUE
C
C IF THERE IS A DUMMY SEWER ASSIGN THE UPSTREAM SEWER
C PROPERTIES TO THE DUMMY SEWER
C TERMINATE THE JOB IF NO DOWNSTREAM SEWER IS FOUND
C
      IUMHN=UMHN(N)
      IF (IUMHN-L) 2190,1940,2190
 2190 CONTINUE
      PVMAX(N)=PVMAX(N-1)
      PDIA(N)=PDIA(N-1)
      PENERGY(N)=PENERGY(N-1)
      PDINV(N)=PDINV(N-1)
      N=N+1
      KC=KC+1
      IF (N-NSTAGE) 2180,2180,2200
 2200 CONTINUE
      WRITE (MO,2210) L
 2210 FORMAT (///5X,'JOB TERMINATED',/5X,'ERROR AT DUMMY SEWER ASSIGNMEN
     1T ON BRANCH = ',I3)
```

```
      REWIND MTP
      RETURN
 2240 CONTINUE
 2250 CONTINUE
C
C
C  O U T P U T   S E C T I O N
C
C
      WRITE (NO,2260)
 2260 FORMAT (1H1,5X,'LINE',1X,'QMAX',2X,'QMIN',1X,'SLOPE',1X,'DIAM',1X,
     1'QFULL',1X,'VFULL',1X,'VMAX',1X,'VMIN',2X,'MAX',/65X,'DEPTH',/12X,
     2'(CFS)',1X,'(CFS)',6X,'(IN)',1X,'(CFS)',1X,'(FPS)',1X,'(FPS)',1X,'
     3(FPS)',1X,'(FT)',/)
      WRITE (NO,2270) (I,PQMAX(I),PQMIN(I),PSLOPE(I),PDIA(I),PQFULL(I),P
     1VFULL(I),PVMAX(I),PVMIN(I),PDEPTH(I),I=1,NLINES)
 2270 FORMAT (I9,2F6.2,F6.4,F5.1,F5.2,F5.1,2F6.1,F6.2)
      WRITE (NO,2280)
 2280 FORMAT (//5X,'LINE',2 ,'MAX',2X,'MANH',1X,'INVERT',1X,'FALL',1X,'U
     1INV',2X,'DINV',2X,'UGELEV',2X,'DGELEV',/10X,'ENERGY',1X,'LOSS',2X,
     2'DROP',2X,'ELEV',2X,'ELEV',/10X,'(FT)',2X,'(FT)',1X,'(FT)',2X,'(FT
     3)',1X,'(FT)',2X,'(FT) ,4X,'(FT)',3X,'(FT)',/)
      WRITE (NO,2290) (I,PENERGY(I),PMHLSS(I),PINVDP(I),PFALL(I),PUINV(I
     1),PDINV(I),UGELEV(I),DGELEV(I),I=1,NLINES)
 2290 FORMAT (I9,F6.2,F7.2,F5.2,F6.2,F9.2,3F7.2)
      DO 2330 I=1,NLINES
      IF (PDROP(I)-0.) 2300,2320,2300
 2300 CONTINUE
      WRITE (NO,2310) I,PDROP(I)
 2310 FORMAT (//,5X,'ON SEWER',I4,3X,'ENERGY DROP (FT) = ',F5.1)
 2320 CONTINUE
 2330 CONTINUE
      WRITE (NO,2340)
 2340 FORMAT (//5X,'LINE',8 ,'COST',/)
      WRITE (NO,2350) (I,PCOST(I),I=1,NLINES)
 2350 FORMAT (I9,F15.2)
      WRITE (NO,2360) TCOST
 2360 FORMAT (/5X,'TOTAL COST = ',F12.2,' DOLLARS')
      WRITE (NO,2370)
 2370 FORMAT (/5X,'UMHN',4X,'COVER',/12X,'(FT)',/)
      DO 2390 I=1,NLINES
      COVER=UGELEV(I)-PUINV(I)-1.1*PDIA(I)/12.
      WRITE (NO,2390) UMHN(I),COVER
 2380 FORMAT (F9.2,F9.1)
 2390 CONTINUE
      WRITE (NO,2400)
 2400 FORMAT (/5X,'LINE',2X 'ELELEV',/12X,'(FT)'/)
      WRITE (NO,2410) (K,ELELEV(K),K=1,NLINES)
 2410 FORMAT (I7,F9.2)
C
C REPOSITION TAPE
C
      IF (NSEWER.LT.3) GO TO 2430
      DO 2420 K=1,KR
      READ (MTP,1460) (IDREST(J),J=1,IRE)
 2420 CONTINUE
 2430 CONTINUE
C
```

```
C IF PROGRAM IS ON THE FIRST PASS
C SET THE CONSTRAINT ELEVATIONS Y FEET ABOVE AND BELOW
C THE ENERGY LINE ELEVATION ELELEV(I)
C CHECK TO ENSURE CONSTRAINT ELEVATIONS DO NOT VIOLATE
C PREVIOUS CONSTRAINT ELEVATIONS
C
      IF (SIGN4-0.) 2460,2440,2460
2440  CONTINUE
      Y = 0.5
      DO 2450 I=1,NLINES
      PWT=(PDIA(I)/12.+1.)/ 2.
      FACT=PWT
      IF (SIGN2.EQ.2.) FACT=PWT+PDIA(I)/12.-DENERGY(I)
      Z=ELELEV(I)+Y
      IF (ELCU(I).EQ.0.) ELCU(I)=Z
      IF (ELCU(I).NE.0..AND.Z.LT.ELCU(I)) ELCU(I)=Z
      Z=DGELEV(I)-COVMIN-FACT
      IF (ELCU(I).GT.Z) ELCU(I)=Z
      Z=ELELEV(I)-Y
      IF (ELCL(I).EQ.0.) ELCL(I)=Z
      IF (ELCL(I).NE.0..AND.Z.GT.ELCL(I)) ELCL(I)=Z
      IF (ELCU(I).LT.ELCL(I )) ELCU(I)=ELCL(I)+Y
      ISTAGE(I)=I
2450  CONTINUE
      SIGN4=4.
      GO TO 590
2460  CONTINUE
      RETURN
      END
```

```
      SUBROUTINE FLOW
C
C COMPUTES THE DESIGN FLOWS QDMAX(II) AND QDMIN(II)
C MINIMUM DOMESTIC FLOW = AVERAGE FLOW
C NO PEAKING FACTORS APPLIED TO THE COMMERCIAL AND INDUSTRIAL
C FLOWS
C
      COMMON CPLF(13,16),CPOX,DCLF(20),SCLF(25),ICOST,NI,NO,NTP,
     1 IUSER(5),IDAY,IMONTH,IYEAR,NPIPES,DIA(20),
     2 ACRES(100),DMHN(100),DIAMAX,DEPTH,DROP,
     3 ENERGY,
     4 FALL,
     5 G,
     6 IUGS(100),IDGS(100),
     7 NLINES,NSTAGE,N,N1,NROW,
     8 PEAKFS,POPD(100),PIPE,PI
      COMMON QADD(100),QDMAX(100),QDMIN(100),
     1 QFULL,
     2 RCOM,RDOM,RIND,RINF,RMANN,RUN(100),
     3 SLOPE,SIGN3,
     4 TYPE(100),
     5 UMHN(100),
     6 VMAX,VMIN,VFULL,VMAXP,VMINP
      DIMENSION POP(100),QCOM(100),QDOM(100),QINF(100),QIND(100)
C
      PKFMAX=5.
      PKFMIN=2.
      PEAKF=0.
C
C CLEAR ARRAYS
C
      DO 10 II=1,NLINES
      QDMAX(II)=0.
      QDMIN(II)=0.
      POP(II)=0.
      QCOM(II)=0.
      QIND(II)=0.
      QINF(II)=0.
10    CONTINUE
      DO 130 II=1,NLINES
      DO 40 J=1,NLINES
C
C CUMULATE UPSTREAM POPULATION AND FLOWS
C
      IF (UMHN(II)-DMHN(J)) 30,20,30
20    CONTINUE
      POP(II)=POP(J)+POP(II
      QCOM(II)=QCOM(J)+QCOM(II)
      QIND(II)=QIND(J)+QIND(II)
      QINF(II)=QINF(J)+QINF(II)
      QADD(II)=QADD(J)+QADD(II)
30    CONTINUE
40    CONTINUE
      XRCOM=RCOM
      XRIND=RIND
C
C SELECT THE FLOW TYPE, DOMESTIC OR COMMERCIAL OR INDUSTRIAL
C
      IF (TYPE(II)-2.) 50,60,70
```

```
50      CONTINUE
        XRCOM=0.
        XRIND=0.
        GO TO 90
60      CONTINUE
        POPD(II)=0.
        XRIND=0.
        GO TO 80
70      CONTINUE
        POPD(II)=0.
        XRCOM=0.
80      CONTINUE
        IF (POPD(II).EQ.0.) GO TO 120
C
C COMPUTE THE DOMESTIC FLOW PEAKING FACTOR, PEAKE
C
        POP(II)=POPD(II)*ACRES(II)+POP(II)
        P1000=POP(II)/1000.
        IF (PEAKES-1.) 90,100,100
90      CONTINUE
        PEAKE=(18.+SQRT(P1000))/(4.+SQRT(P1000))
        GO TO 110
100     CONTINUE
        PEAKE=5./P1000**0.2
110     CONTINUE
        IF (PEAKE.GT.PKEMAX) PEAKE=PKEMAX
        IF (PEAKE.LT.PKEMIN) PEAKE=PKEMIN
120     CONTINUE
C
+ COMPUTE THE FLOWS
C
        QDOM(II)=RDOM*POP(II)*PEAKE
        QCOM(II)=XRCOM*ACRES(II)+QCOM(II)
        QIND(II)=XRIND*ACRES(II)+QIND(II)
        QINF(II)=RINF*ACRES(II)+QINF(II)
        QDOM(II)=QDOM(II)*1.8566E-6
        QCMCFS=QCOM(II)*1.8566E-6
        QIDCFS=QIND(II)*1.8566E-6
        QIFCFS=QINF(II)*1.8566E-6
        QDMAX(II)=QDOM(II)+QCMCFS+QIDCFS+QIFCFS+QADD(II)
        QDMIN(II)=QDOM(II)/PEAKE+QCMCFS+QIDCFS+QIFCFS+QADD(II)
130     CONTINUE
        RETURN
        END
```

```
      FUNCTION PIPE1(KN,KX,SDIA)
C
C DESIGN BASED ON FULL FLOW CONDITION
C SELECTS THE SMALLEST PIPE TO MEET THE REQUIREMENT THAT
C SEWER CAPACITY, QFULL, IS GREATER THAN QDMAX(I) FOR GIVEN HEAD
C WITHOUT VIOLATION OF THE VELOCITY CONSTRAINTS
C MAXIMUM VELOCITY, VMAXP = VFULL, COMPUTED BY KUTTERS FORMULA
C POMEROYS FORMULA IS USED TO COMPUTE MINIMUM VELOCITY, VMINP
C FLOW RATE BY MANNINGS FORMULA
C POMSK = POMEROYS CONSTANT K
C PIPE1 = DIA(NPIPES) BY DEFAULT
C
      COMMON CPLE(13,15),CPOX,PCLE(20),SCLE(25),ICOST,NI,NO,NTP,
     1 IUSER(5),IDAY,IMONTH,IYEAR,NPIPES,DIA(20),
     2 ACRES(100),DUHN(100),DIAMAX,DEPTH,DROP,
     3 ENERGY,
     4 FALL,
     5 G,
     6 IUGS(100),IDGS(100),
     7 NLINES,NSTAGE,N,N1,NROW,
     8 PEAKES,PORD(100),PIPE,PI
      COMMON QADD(100),QDMAX(100),QDMIN(100),
     1 QFULL,
     2 RCOM,RDOM,RIND,RINF,RMANN,RUN(100),
     3 SLOPE,SIGN3,
     4 TYPE(100),
     5 UMHN(100),
     6 VMAX,VMIN,VFULL,VMAXP,VMINP
C
      COFFE=0.467
      DROP=0.
      POMSK=-956.*RMANN+29.66
      FALL=FLOAT(KX)/100.
      SLOPE=FALL/RUN(KN)
      D=SDIA
10    CONTINUE
20    CONTINUE
C
C COMPUTE SEWER CAPACITY, QFULL, BY MANNINGS EQUATION
C
      R=D/4.
      A=(PI*D**2)/4.
      QFULL=COFFE*A*(G*R*R)**0.3333*SQRT(SLOPE)/RMANN
      IF (QFULL-QDMAX(KN)) 40,30,30
30    CONTINUE
C
C COMPUTE VFULL BY KUTTERS EQUATION
C
      CNUM=41.65+0.00281/SLOPE+1.811/RMANN
      CDENOM=1.+RMANN*(41.6 +0.00281/SLOPE)/SQRT(D/4.)
      C=CNUM/CDENOM
      VFULL=C*SQRT((D/4.)*SLOPE)
      IF (VFULL.LT.VMIN) GO TO 40
      IF (VFULL.GT.VMAX) GO TO 130
      PIPE1=D
      GO TO 120
C
C INCREMENT DIAMETER D
C
```

```
40     CONTINUE
50     CONTINUE
60     CONTINUE
       IF (D-0.68) 70,80,80
70     CONTINUE
       D=10./12.
       GO TO 110
80     CONTINUE
       IF (D-0.84) 90,100,100
90     CONTINUE
       D=1.
       GO TO 110
100    CONTINUE
       D=D+0.25
110    CONTINUE
       IF (D.LT.DIAMAX) GO TO 10
C
C IF D IS GREATER THAN THE OUTFALL DIAMETER
C SET PIPE1 = DEFAULT PIPE SIZE
C
       PIPE1=DIA(NPIPES)/12.
120    CONTINUE
       VMINP=1.4*ROMSK*(SLOPE**0.41)*(QDMIN(KN))**0.24
       VMAXP=VFULL
       DEPTH=0.
       ENERGY=0.
       RETURN
C
130    CONTINUE
C
C DROP MANHOLE ROUTINE
C
C COMPUTE MINIMUM AND MAXIMUM SLOPES SMIN AND SMAX
C FROM MANNINGS EQUATION
C
       SMIN=(VMIN*RMANN)/(COEFF*(G*R*R)**0.3333)
       SMIN=SMIN*SMIN
       SMAX=SMIN*(VMAX/VMIN)**2
       QFULL=A*VMAX
       S=SMAX
140    CONTINUE
       IF (QFULL.LT.QDMAX(KN)) GO TO 50
       CNUM=41.65+0.00281/S+1.811/RMANN
       CDENOM=1.+RMANN*(41.6 +0.00281/S)/SQRT(D/4.)
       C=CNUM/CDENOM
       VFULL=C*SQRT((D/4.)*S)
       IF (VFULL.GT.VMAX) GO TO 50
       HEAD=S*RUN(KN)
       IF (HEAD-FALL) 150,160,160
150    CONTINUE
       DROP=FALL-HEAD
       SLOPE=S
       GO TO 20
160    CONTINUE
       S=S-0.1*(SMAX-SMIN)
       IF (S.LT.SMIN) GO TO 60
       QFULL=COEFF*A*(G*R*R) *0.3333*SQRT(S)/RMANN
       GO TO 140
       END
```

```
      FUNCTION PIPE2(KN,KX, DIA)
C
C SELECTS THE SMALLEST PIPE TO MEET THE REQUIREMENT THAT
C SEWER CAPACITY, QFULL, EXCEEDS QDMAX(I) FOR GIVEN HEAD
C PARTIAL FLOW CONDITIONS ARE THEN ANALYSED
C FLOW RATE AND MAXIMUM VELOCITY BY MANNINGS EQUATION
C AND HYDRAULIC ELEMENTS
C MANNINGS ROUGHNESS COEFFICIENT, RMANN, VARIABLE WITH DEPTH
C MAXIMUM SEWER CAPACITY OCCURS AT DRATIO = 0.9
C DRATIO = DEPTH OF FLOW/DIAMETER OF SEWER
C QRATIO = QDMAX(KN)/QFULL, FOR GIVEN SLOPE
C VRATIO = VELOCITY AT PARTIAL FLOW/VELOCITY AT FULL FLOW
C FOR GIVEN SLOPE
C QS = DESIGN FLOW, QDMAX(KN)
C QF = CAPACITY OF SEWER FLOWING FULL AT VELOCITY = VMIN
C QSOVQF = QS/QF
C SF = SLOPE OF SEWER FLOWING FULL AT VELOCITY = VMIN
C SC = SLOPE OF SEWER CARRYING QS AND AS SELF CLEANING
C AS SEWER FLOWING FULL WITH VELOCITY = VMIN
C SCOVSF = SC/SF
C POMEROYS FORMULA IS USED TO COMPUTE THE MINIMUM VELOCITY
C POMSK = POMEROYS CONSTANT K
C PIPE2 = DIA(NPIPES) BY DEFAULT
C
      COMMON CPLF(13,15),CROX,DCLF(20),SCLF(25),ICOST,NI,NO,NTP,
     1 IUSER(5),IDAY,IMONTH,IYEAR,NPIPES,DIA(20),
     2 ACRES(100),DMHN(100),DIAMAX,DEPTH,DROP,
     3 ENERGY,
     4 FALL,
     5 G,
     6 IUGS(100),IDGS(100),
     7 NLINES,NSTAGE,N,N1,NROW,
     8 PEAKES,PCPD(100),PIPE,PI
      COMMON QADD(100),QDMAX(100),QDMIN(100),
     1 QFULL,
     2 RCOM,RDOM,RIND,RINF,RMANN,RUN(100),
     3 SLOPE,SIGN3,
     4 TYPE(100),
     5 UMHN(100),
     6 VMAX,VMIN,VFULL,VMAXP,VMINP
C
      COEFF=0.467
      POMSK=-956.*RMANN+29.66
      DROP=0.
      FALL=FLOAT(KX)/100.
      SLOPE=FALL/RUN(KN)
      D=SDIA
10    CONTINUE
C
C COMPUTE THE SEWER CAPACITY, QFULL, BY MANNINGS FORMULA
C
      R=D/4.
      A=(PI*D**2)/4.
      QFULL=COEFF*A*(G*R*R) *0.3333*SQRT(SLOPE)/RMANN
      IF (QFULL-QDMAX(KN)) 20,110,110
C
C INCREMENT DIAMETER D
C
20    CONTINUE
```

```
30     CONTINUE
40     CONTINUE
50     CONTINUE
       IF (D-0.68) 60,60,70
60     CONTINUE
       D=10./12.
       GO TO 100
70     CONTINUE
       IF (D-0.84) 80,90,90
80     CONTINUE
       D=1.
       GO TO 100
90     CONTINUE
       D=D+0.25
100    CONTINUE
       IF (D.LT.DIAMAX) GO TO 10
C
C IF D IS GREATER THAN THE OUTFALL DIAMETER
C SET PIPE2 = DEFAULT PIPE SIZE
C
       PIPE2=DIA(NPIPES)/12.
       VMINP=0.
       VMAXP=0.
       DEPTH=DIA(NPIPES)/12.
       ENERGY=DIA(NPIPES)/12
       RETURN
C
110    CONTINUE
C
C COMPUTE MAXIMUM VELOCITY VMAXP
C CHECK IF VMAXP IS GREATER THAN VMAX
C
       VFULL=QFULL/A
       QRATIO=QDMAX(KN)/QFULL
       IF (QRATIO-0.14) 130,140,140
130    CONTINUE
       DRATIO=6.23*QRATIO-59.23*QRATIO**2+214.44*QRATIO**3
       GO TO 150
140    CONTINUE
       DRATIO=0.2+0.73*QRATIO-.04*QRATIO**2
150    CONTINUE
       VRATIO=.018+3.47*DRATIO-7.27*DRATIO**2+9.04*DRATIO**3-4.28*DRATIO*
      1*4
       VMAXP=VFULL*VRATIO
       IF (VMAXP.GT.VMAX) GO TO 210
C
C CHECK THAT SEWER SLOPE IS GREATER THAN SLOPE FOR EQUAL
C SELF CLEANSING, SC
C
       SF=(VMIN*RMANN)/(COEFF*(G*R*R)**0.3333)
       SF=SF*SF
       QF=A*VMIN
       QSOVQF=QDMAX(KN)/QF
       IF (QSOVQF-0.19) 160,170,170
160    CONTINUE
       SCOVSF=9.79-269.29*QSOVQF+3540.58*QSOVQF**2-20348.2*QSOVQF**3+4187
      12.5*QSOVQF**4
       GO TO 190
```

```
170   CONTINUE
      IF (QSOVQF-1.) 180,200,200
180.  CONTINUE
      SCOVSF=2.22-6.26*QSOVQF+12.46*QSOVQF**2-13.19*QSOVQF**3+5.74*QSOVQ
     1F**4
190   CONTINUE
      SC=SCOVSF*SF
      IF (SLOPE.LT.SC) GO TO 30
200   CONTINUE
205   CONTINUE
      PIPE2=D
      DEPTH=DRATIO*D
      VMINP=1.4*POWSK*(SLOPE**0.41)*(QDMIN(KN1)**0.24
      ENERGY=DEPTH+VMAXP**2/(2.*G)
      RETURN
210   CONTINUE
C
C DROP MANHOLE ROUTINE
C
C COMPUTE THE MINIMUM AND MAXIMUM SLOPES, SMIN AND SMAX
C FROM MANNINGS EQUATION ASSUMING FULL FLOW
C
      SMIN=(VMIN*RMANN)/(COEFF*(G*R*R)**0.3333)
      SMIN=SMIN*SMIN
      VFULL=VMAX
      SMAX=SMIN*(VFULL/VMIN)**2
      QFULL=A*VFULL
      S=SMAX
220   CONTINUE
      IF (QFULL.LT.QDMAX(KN)) GO TO 40
      HEAD=S*RUN(KN)
      IF (HEAD-FALL) 230,270,270
230   CONTINUE
      QRATIO=QDMAX(KN)/QFULL
      IF (QRATIO-0.14) 240,250,250
240   CONTINUE
      DRATIO=6.23*QRATIO-59.23*QRATIO**2+214.44*QRATIO**3
      GO TO 260
250   CONTINUE
      DRATIO=0.2+0.73*QRATIO-.04*QRATIO**2
260   CONTINUE
      VRATIO=.018+3.47*DRATIO-7.27*DRATIO**2+9.04*DRATIO**3-4.28*DRATIO*
     1*4
      VMAXP=VRATIO*VFULL
      IF (VMAXP.GT.VMAX) GO TO 280
      DROP=FALL-HEAD
      SLOPE=S
      GO TO 205
270   CONTINUE
280   CONTINUE
      S=S-0.1*(SMAX-SMIN)
      IF (S.LT.SMIN) GO TO 50
      VFULL=COEFF*(G*R*R)**.3333*SQRT(S)/RMANN
      QFULL=A*VFULL
      GO TO 220
      END
```

```
      FUNCTION CSTPM(KN,KS,KX)
C
C COMPUTES THE COST OF THE (PIPE + MANHOLES)
C COST MAY BE COMPUTED FROM COST ARRAY OR FROM COST FUNCTION
C MAX PIPE DIAMETER = 42 INCH
C MINIMUM CUT = 6 FEET
C MAXIMUM AVERAGE CUT = 28 FEET
C IF KN = 1 , UPSTREAM MANHOLE COST INCLUDED
C IF KN = NSTAGE , DOWNSTREAM MANHOLE COST NOT INCLUDED
C UCUT = UPSTREAM CUT
C DCUT = DOWNSTREAM CUT
C CPLF(PIPE,CUT) = COST OF THE PIPE PER LINEAL FOOT
C DIST = VERTICAL DISTANCE FROM THE ENERGY LINE
C TO THE TRENCH BOTTOM
C MINIMUM CLEARANCE TO SUBGRADE, CLRNCE = 4 INCHES
C
      COMMON CPLF(13,15),CBOX,DCLF(20),SCLF(25),ICOST,NI,NO,NTP,
     1 IUSER(5),IDAY,IMONTH,IYEAR,NPIPES,DIA(20),
     2 ACRES(100),DMHN(100),DIAMAX,DEPTH,DROP,
     3 ENERGY,
     4 FALL,
     5 G,
     6 IUGS(100),IDGS(100),
     7 NLINES,NSTAGE,M,NI,NROW,
     8 PEAKFS,POPD(100),PIPE,PI
      COMMON QADD(100),QDMAX(100),QDMIN(100),
     1 QFULL,
     2 RCOM,RDOM,RIND,RINF,RMANN,RUN(100),
     3 SLOPE,SIGN3,
     4 TYPE(100),
     5 UMHN(100),
     6 VMAX,VMIN,VFULL,VMAXP,VMINP
C
      PWT=(PIPE+1.)/12.
      CLRNCE=0.33
      DIST=PIPE+PWT+CLRNCE
      IF (SIGN3.EQ.3.) DIST=ENERGY+PWT+CLRNCE
C
C COMPUTE AVERAGE CUT
C
      IF (KS-KX) 10,20,20
10    CONTINUE
      UCUT=6.
      GO TO 30
20    CONTINUE
      IUCUT=-(IUGS(KN)-(KS-KX))
      UCUT=FLOAT(IUCUT)
      UCUT=UCUT/100.+DIST+DROP
      IF (UCUT.LT.6.) UCUT=6.
30    CONTINUE
      IDCUT=-(IDGS(KN)-KS)
      DCUT=FLOAT(IDCUT)
      DCUT=DCUT/100.+DIST
      AVCUT=(UCUT+DCUT)/2.
      AAVCUT=AVCUT+0.5
      IAVCUT=AAVCUT
C
      IF (ICOST.EQ.1) GO TO 40
C
```

```
C COMPUTE THE COSTS FROM THE COST ARRAY
C
C LOCATE COLUMN OF THE COST ARRAY
C
      IF (IAVCUT.EQ.6) NCOL=1
      IF (IAVCUT.EQ.7) NCOL=2
      IF (IAVCUT.EQ.8) NCOL=3
      IF (IAVCUT.EQ.9) NCOL=4
      IF (IAVCUT.EQ.10) NCOL=5
      IF (IAVCUT.EQ.11) NCOL=6
      IF (IAVCUT.EQ.12) NCOL=7
      IF (IAVCUT.GT.12.AND.IAVCUT.LE.14) NCOL=8
      IF (IAVCUT.GT.14.AND.IAVCUT.LE.16) NCOL=9
      IF (IAVCUT.GT.16.AND.IAVCUT.LE.18) NCOL=10
      IF (IAVCUT.GT.18.AND.IAVCUT.LE.20) NCOL=11
      IF (IAVCUT.GT.20.AND.IAVCUT.LE.22) NCOL=12
      IF (IAVCUT.GT.22.AND.IAVCUT.LE.24) NCOL=13
      IF (IAVCUT.GT.24.AND.IAVCUT.LE.26) NCOL=14
      IF (IAVCUT.GT.26) NCOL=15
      PPIPE=PIPE*12.
      IPIPE=PPIPE
C
C LOCATE ROW OF THE COST ARRAY
C
      IF (IPIPE.LE.9) NROW=1
      IF (IPIPE.GT.9.AND.IPIPE.LE.11) NROW=2
      IF (IPIPE.GT.11.AND.IPIPE.LE.13) NROW=3
      IF (IPIPE.GT.13.AND.IPIPE.LE.16) NROW=4
      IF (IPIPE.GT.16.AND.IPIPE.LE.19) NROW=5
      IF (IPIPE.GT.19.AND.IPIPE.LE.22) NROW=6
      IF (IPIPE.GT.22.AND.IPIPE.LE.25) NROW=7
      IF (IPIPE.GT.25.AND.IPIPE.LE.28) NROW=8
      IF (IPIPE.GT.28.AND.IPIPE.LE.31) NROW=9
      IF (IPIPE.GT.31.AND.IPIPE.LE.34) NROW=10
      IF (IPIPE.GT.34.AND.IPIPE.LE.40) NROW=11
      IF (IPIPE.GT.37.AND.IPIPE.LE.40) NROW=12
      IF (IPIPE.GT.40.AND.IPIPE.LE.43) NROW=13
      COSTP=CPLF(NROW,NCOL)*RUN(KN)
      CSTM=COSTM(KN,UCUT,DCUT)
      CSTPM=COSTP+CSTM
      RETURN
C
40    CONTINUE
C
C COMPUTE THE COSTS FROM COST FUNCTIONS
C
      XRUN=RUN(KN)
      COSTP=PRICEP(AVCUT,PIPE,XRUN)
      CSTM=COSTM(KN,UCUT,DCUT)
      CSTPM=COSTP+CSTM
      RETURN
      END
```

```
      FUNCTION COSTM(KKN,UUCUT,DDCUT)
C
C COMPUTES MANHOLE COSTS
C SHAFT COSTS MAY BE COMPUTED
C FROM COST ARRAY OR FROM COST FUNCTION
C MANHOLE COST = BOX COST + SHAFT COST
C              = CBOX + SCLF(ISHAFT)
C CUMH = COST OF THE UPSTREAM MANHOLE
C CDMH = COST OF THE DOWNSTREAM MANHOLE
C MANHOLE BOX HEIGHT = BOXHT
C CSHAFT = COST OF THE MANHOLE SHAFT
C SHAFTH = SHAFT HEIGHT
C SCLF(25) = SHAFT COST
C COSTDP = ADDITIONAL COST FOR THE DROP
C        = ADDITIONAL MANHOLE COST + DROP PIPE COST
C        = AMHC + DCOST
C DCLF(NROW) = DROP COST PER LINEAL FOOT FOR PIPE NROW
C
      COMMON CPLF(13,15),CBOX,DCLF(20),SCLF(25),ICOST,NI,NO,NTP,
     1 IUSER(5),IDAY,IMONTH,IYEAR,NPIPES,DIA(20),
     2 ACRES(100),DMHN(100),DIAMAX,DEPTH,DROP,
     3 ENERGY,
     4 FALL,
     5 G,
     6 IUGS(100),IDGS(100),
     7 NLINES,NSTAGE,N,NI,NROW,
     8 PEAKFS,POPD(100),PIPE,PI
      COMMON QADD(100),QDMAX(100),QDMIN(100),
     1 QFULL,
     2 RCOM,RDOM,RIND,RINF,RMANN,RUN(100),
     3 SLOPE,SIGN3,
     4 TYPE(100),
     5 UMHN(100),
     6 VMAX,VMIN,VFULL,VMAXP,VMINP
C
      CUMH=0.
      CDMH=0.
      COSTDP=0.
      BOXHT=5.0
C
      IF (ICOST.EQ.1) GO TO 60
C
C COMPUTE THE COST FROM THE COST ARRAYS
C
      IF (KKN-NSTAGE) 10,20,10
10    CONTINUE
      SHAFTH=DDCUT-BOXHT
      HSHAFT=SHAFTH+0.5
      ISHAFT=HSHAFT
      CSHAFT=SCLF(ISHAFT)
      CDMH=CBOX+CSHAFT
20    CONTINUE
      IF (KKN-NI) 40,30,40
30    CONTINUE
      SHAFTH=UUCUT-BOXHT
      HSHAFT=SHAFTH+0.5
      ISHAFT=HSHAFT
      CSHAFT=SCLF(ISHAFT)
      CUMH=CBOX+CSHAFT
```

```
40      CONTINUE
        IF (DROP.EQ.0.) GO TO 50
C
C COSTDP ROUTINE
C OCUT = ORIGINAL CUT OR CUT CONSIDERED FOR
C THE DOWNSTREAM MANHOLE FOR THE PREVIOUS STAGE
C COSTN = PRESENT COST OF UPSTREAM MANHOLE SHAFT
C OCOST = ORIGINAL COST OF UPSTREAM MANHOLE SHAFT
C
        OCUT=UUCUT-DROP
        OSHFTH=OCUT-BOXHT
        OHT=OSHFTH+0.5
        IOHT=OHT
        OCOST=SCLF(IOHT)
        SHFTHN=UUCUT-BOXHT
        IF (SHFTHN.GT.25.) SHFTHN=25.
        HTN=SHFTHN+0.5
        IHTN=HTN
        COSTN=SCLF(IHTN)
        AMHC=COSTN-OCOST
        DCOST=DCLF(NROW)*DROP
        IF (DROP.LE.2.0) DCOST=0.
        IF (KKN.EQ.N1) DCOST=0.
        COSTDP=AMHC+DCOST
50      CONTINUE
        COSTM=CUMH+CDMH+COSTDP
        RETURN
C
60      CONTINUE
C
C COMPUTE THE COST FROM COST FUNCTIONS
C
        IF (KKN-NSTAGE) 70,80,70
70      CONTINUE
        SHAFTH=DDCUT-BOXHT
        CSHAFT=PRICES(SHAFTH)
        CDMH=CBOX+CSHAFT
80      CONTINUE
        IF (KKN-N1) 100,90,100
90      CONTINUE
        SHAFTH=UUCUT-BOXHT
        CSHAFT=PRICES(SHAFTH)
        CUMH=CBOX+CSHAFT
100     CONTINUE
        IF (DROP.EQ.0.) GO TO 110
C
C COSTDP ROUTINE
C
        OCUT=UUCUT-DROP
        OSHFTH=OCUT-BOXHT
        OCOST=PRICES(OSHFTH)
        SHFTHN=UUCUT-BOXHT
        IF (SHFTHN.GT.25.) SHFTHN=25.
        COSTN=PRICES(SHFTHN)
        AMHC=COSTN-OCOST
        DCOST=PRICED(DROP,PIPE)
        IF (DROP.LE.2.) DCOST=0.
        IF (KKN.EQ.N1) DCOST=0.
```

```
      COSTDP=AMHC+DCOST
110   CONTINUE
      COSTM=CUMH+CPMH+COSTDP
      RETURN
      END
```

APPENDIX 2

List of Program Variables

Variable	Description
A	cross-sectional area of pipe
AAVCUT	average depth to subgrade
ACRES	array of incremental drainage areas in acres
AMHC	additional manhole cost for a drop
AVCUT	average depth to subgrade
BOXHT	height of manhole chamber
C	Kutter's coefficient
CBOX	cost of manhole chamber
CDMH	cost of downstream manhole
CLRNCE	clearance for underside of pipe to subgrade
COEFF	coefficient in Manning's equation
COST	cost of pipe and manholes for a stage
COSTDP	additional cost for a drop
COSTJ	array of branch costs
COSTN	present cost of a manhole
COSTP	cost of a sewer for a stage
COVER	depth of cover over a sewer
COVMAX	maximum cover requirement

Variable	Description
COVMIN	minimum cover requirement
CPLF	array of pipe costs
CSHAFT	cost of manhole shaft
CSTM	cost of manholes for a stage
CSTPM	cost of pipe and manholes for a stage
CUMH	cost of upstream manhole
CUT	average depth to subgrade
D	pipe diameter
DCLF	array of drop pipe costs
DCOST	drop pipe cost
DDCUT	downstream cut
DCUT	downstream cut
DEPTH	depth of flow in the sewer
DGELEV	array of downstream ground elevations
DGS	downstream ground elevation
DIA	array of pipe sizes
DIAMAX	maximum allowable diameter
DIAMIN	minimum allowable diameter
DIST	vertical distance from energy line to subgrade
DMHN	array of downstream manhole numbers
DRATIO	relative depth of flow in a sewer

Variable	Description
DROP	drop manhole height
DTEMP	array of pipe sizes for a stage. Values assigned on completion of design for the stage.
ELCL	array of lower elevation constraints
ELCU	array of upper elevation constraints
ELELEV	array of energy line elevations
ELEVCL	lower state constraint
ELEVCU	upper state constraint
ELOSS	manhole energy loss
ENERGY	specific energy
FACT	vertical distance from the energy line to the outside top of the sewer
FALL	vertical drop in energy line elevation over the length of the sewer
FNSXN	cumulated cost
FSBESL	optimum system cost
FSBEST	array of best cumulated costs. Assigned during designed of a stage.
FSOPT	array of best cumulated costs. Assigned on completion of design for a stage.
FSTAR	contributory upstream cost

Variable	Description
G	acceleration of gravity
HEAD	SLOPE* RUN for drop pipe stage
HEIGHT	height of drop
HD	maximum decision
HS	maximum state
HSHAFT	manhole shaft height
HTN	shaft height now
I	index variable, identification number for a state
IAVCUT	average cut
IBF	constant
IBF1	constant
IBLANK	constant
ICOL	index variable
ICOST	cost data selector
IDAY	day of the month
IDBESL	identification number for the optimum outfall state
IDBEST	array of best decisions
IDCUT	downstream cut
IDGS	array of downstream ground state values
IDIA	pipe diameter
IDMHN	integer part of DMHN

Variable	Description
IDS	integer part of DGS
IHTN	integer part of HTN
IELCL	array of lower state constraints
IELCU	array of upper state constraints
IELL	integer part of ELEVCL
IELU	integer part of ELEVCU
II	index variable
IMAX	identification number for maximum outfall state
IMIN	identification number for minimum outfall state
IMONTH	month
IOHT	integer part of OHT
INCRMT	state and decision increment
INMK	integer part of UMHN (NMK)
IOPT3	signal for optional printout
IOUT	integer part of OUT
IPIPE	pipe diameter
IREM	problem identification
IROW	index variable
ISHAFT	shaft height
ISTAGE	array of elevation constraint signals
ISUM	cumulated decisions

Variable	Description
ITIED	signal for tied solution
ITIED1	signal for tied solution
ITIED2	signal for tied solution
IUCUT	integer part of UCUT
IUGS	array of upstream ground state values
IUMHN	integer part of UMHN
IUP	identification number for upstream state
IUS	integer part of UGS
IUSER	user's name
IX	best decision
IYEAR	year
IYES	signal
J	index variable
JL	array of lowest feasible upstream states
JNS	array of maximum optimum states at junctions
JS	state variable
JU	array of highest feasible upstream states
JX	decision variable
J34	constant
K	index variable
KB	(number stages stored on tape - 2) *KTAPE

Variable	Description
KBRNCH	array of number of sewers in each branch
KC	index variable
KCL	integer part of ELEVCU
KDEC	array of optimum decisions
KDECK	decision
KK	index variable
KKN	stage
KN	stage
KOUNT1	signal to check if outfall sewer has been selected
KOUNT2	signal to select first feasible pipe in any stage
KOUNT3	signal to show that there is a feasible pipe
KS	state
KSER	number of serial systems
KSERP1	KSER +1
KSS	identification number
KTAPE	number of blocks of data stored on tape for each stage
KTIES	tied solution identification array
KX	decision
K34	index variable
L	index variable
LARGE	number of serial systems

Variable	Description
LAYER	index variable
LAYMAX	index variable
LINES	maximum number of sewers +1
LL	branch identification number
LLHIGH	branch with highest most upstream ground elevation
LM1	LINES -1
LOW	lowest state value for the sewer system
LOWD	low decision value
LOWEST	lowest state value for any branch other than branch LLHIGH
M	index variable
MAXD	maximum decision
MAXS	maximum state for the problem
MFLAG	signal to note tied solutions
M2	index variable
N	stage
NBRNCH	number of branches permited
NCOL	index variable
NCUT	index variable
NCONST	signal for constraint elevations
ND	decision variable
NI	input reference

Variable	Description
NJMAX	maximum number of junctions permitted
NJUNC	junction number
NJUNCS	number of junctions in the problem
NLINES	number of lines in the problem
NMK	N - K
NNS	identification number for the maximum state value in the problem
NO	output reference
NPIPES	number of pipe sizes considered
NPROB	number of problems
NROW	index variable
NRUN	run number
NS	state variable
NSEWER	number of sewers, including dummy sewers, for which decisions have been stored on the scratch tape
NSHAFT	index variable
NSMAX	maximum number of state values permitted
NSMND	NS - ND
NSTAGE	outlet stage for any branch
NSTMN1	NSTAGE - N1
NSTM1	NSTAGE - 1

Variable	Description
NSUP	upstream state
NTP	scratch tape reference
NTY	index variable for tied solutions
N1	furthest upstream stage, in any branch
N2	stage
N3	NSTAGE + 1
OCOST	original cost
OCUT	original cut
OHT	original height
OSHFTH	original shaft height
OUT	- (OUTFAL - X)
OUTDIA	outfall diameter
PCOST	array of the stage costs
PDEPTH	array of the depths of flow
PDIA	array of pipe sizes
PDINV	array of downstream invert elevations
PDROP	array of drop heights
PEAKF	peaking factor
PEAKFS	peaking factor selector
PENERGY	array of specific energies
PFALL	array of FALL values

Variable	Description
PI	constant pi
PINVDP	array of invert elevation drops across a sewer
PIPE	pipe diameter
PIPEJ	array of branch outlet pipe sizes
PKFMAX	maximum peaking factor
PLNGTH	pipe length
PMHLOSS	array of manhole energy losses
POMSK	Pomeroy's coefficient
POP	array of populations
POPD	array of population densities
PQFULL	array of sewer capacities
PPIPE	pipe diameter
PSLOPE	array of pipe slopes
PUINV	array of upstream invert elevations
PVFULL	array of full flow velocities
PVMAX	array of maximum velocities
PVMIN	array of minimum velocities
PWT	pipe wall thickness
PWTMIN	minimum pipe wall thickness
P1000	population in thousands
QADD	array of additional flows

Variable	Description
QCMCFS	commercial flow
QCOM	commercial flow rate
QDMAX	array of maximum design flows
QDMIN	array of minimum design flows
QDOM	domestic flow rate
QF	sewer capacity at minimum velocity VMIN
QFULL	sewer capacity
QIDCFS	industrial flow
QIFCFS	infiltration
QIND	industrial flow rate
QINF	infiltration flow rate
QRATIO	QDMAX/QFULL
QSOVQF	QDMAX/QF
R	hydraulic radius
RCOM	commercial flow rate
RDOM	domestic flow rate
RIND	industrial flow rate
RINF	infiltration flow rate
RMANN	Manning's roughness coefficient
RUN	run number

Variable	Description
S	slope of sewer
SC	slope for equal self cleansing
SCLF	array of manhole shaft costs
SCOVSF	SC/SF
SDIA	smallest permissible diameter
SDIAJ	smallest permissible branch diameter
SDIAM	array of smallest permissible diameters, assigned during design of the stage
SDIAMJ	array of sorted branch outlet diameters
SF	slope of sewer flowing full at velocity, VMIN
SHAFTH	shaft height
SHFTHN	shaft height now
SIGN3	signal to select full flow or partial flow analysis
SIGN4	signal to select first or second pass
SLOPE	slope of sewer energy line
SMALL	smallest cost at a junction
SMALLP	smallest pipe at a junction
SMAX	maximum slope
SMIN	minimum slope
TCOST	cumulated cost of the sewer system
TYPE	variable to select class of flow

Variable	Description
UCUT	upstream cut
UGELEV	array of upstream ground elevations
UGHIGH	highest upstream ground elevation for a stage N1 sewer
UGS	- (UGELEV (1) - X)
UMHN	array of upstream manhole numbers
UNITP	price per lineal foot
UUCUT	upstream cut
VFULL	full flow velocity
VL	maximum downstream velocity
VMAX	maximum velocity permitted
VMAXP	maximum velocity
VMIN	minimum velocity permitted
VMINP	minimum velocity
VRATIO	VMAXP/VFULL
VU	maximum upstream velocity
X	vertical distance between UGHIGH and energy line elevation
XHS	maximum state value
XJN	array of junction numbers
XLOW	low state value
XMAXD	maximum decision value

Variable	Description
XNS	minimum state value
XRCOM	commercial flow rate
XRIND	industrial flow rate
XRUN	length of sewer
Y	value used to compute feasible region in second pass
Z	variable

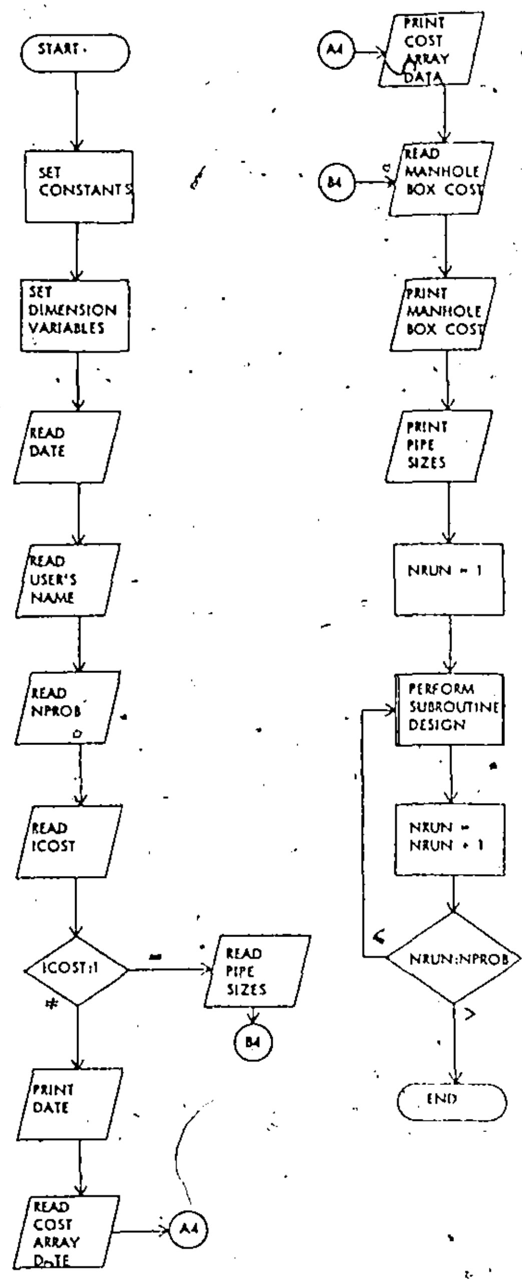
APPENDIX 3

Flow Charts

Name	Page
SEWER	195
DESIGN (Outline)	196
DESIGN	197
FLOW	204
PIPE1	205
PIPE2	206
CSTPM	208
COSTM	209

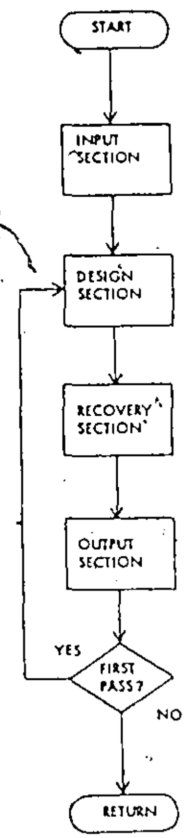
IBM Flowcharting Worksheet

Programmer: W. M. MAIN Program No.: Date: Page: 1 of 1
Chart ID: Chart Name: Program Name: SEWER



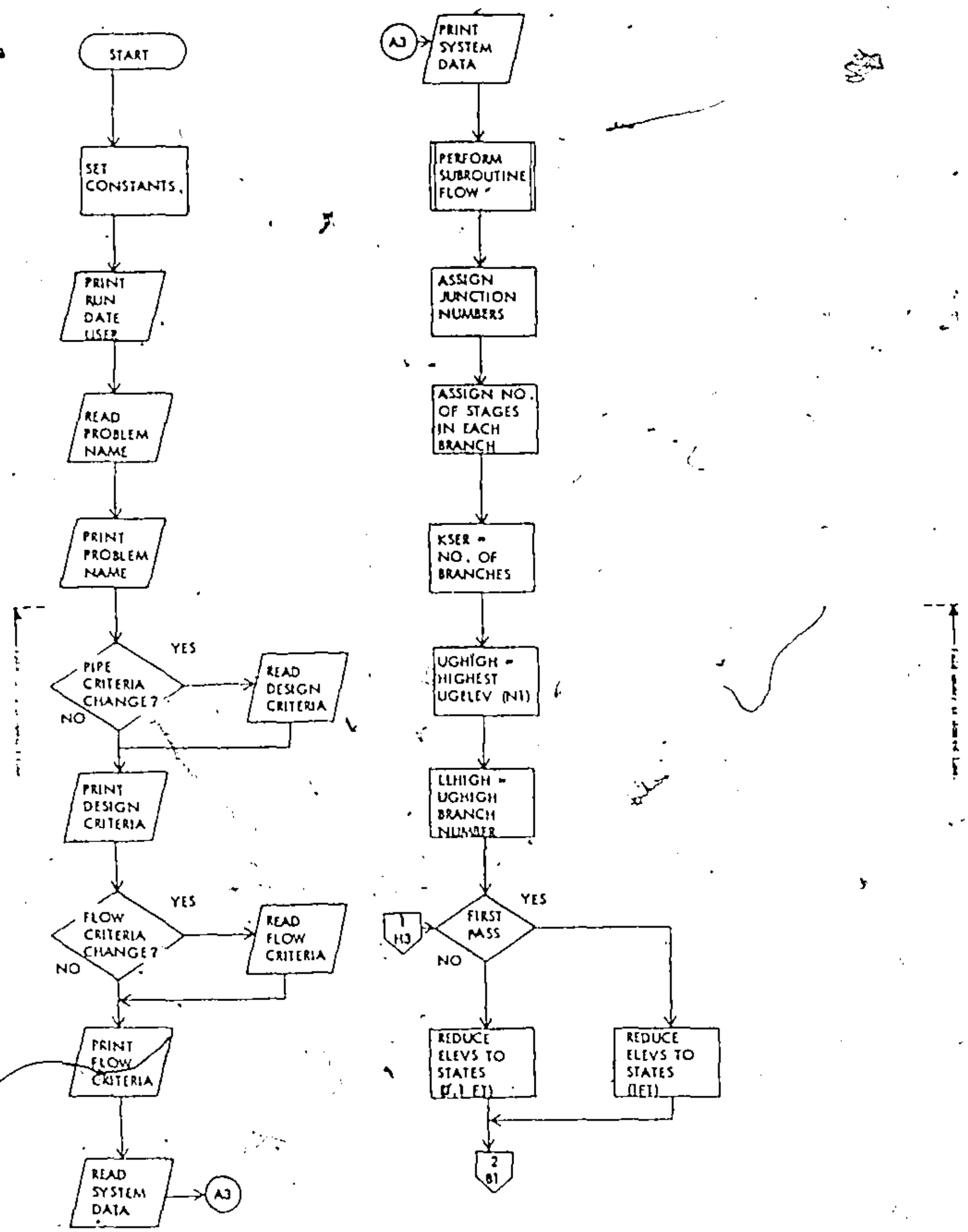
IBM Flowcharting Worksheet

Programmer: W. M. MAIN Program No.: Date: Page:
Chart ID: Chart Name: Program Name: DESIGN (OUTLINE)



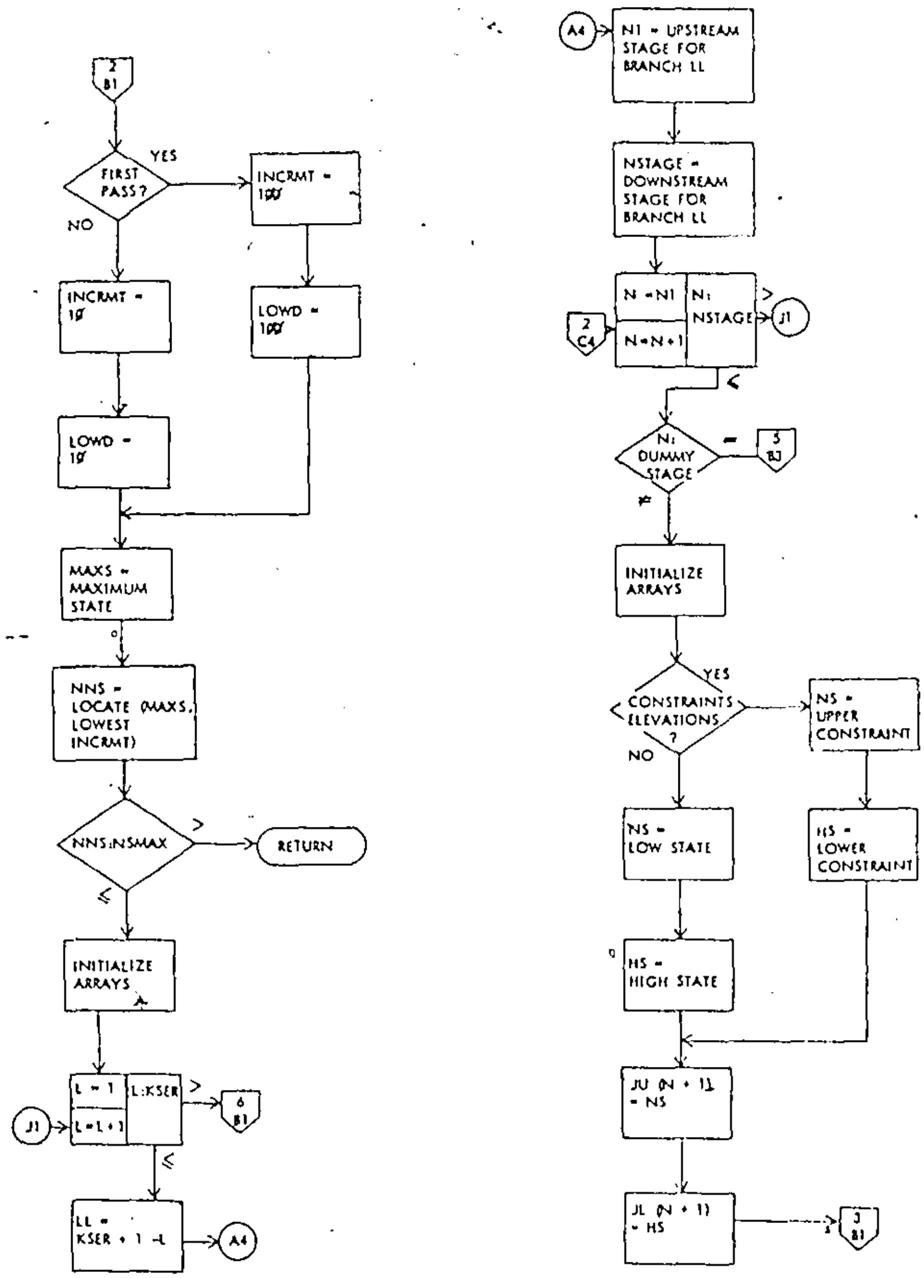
IBM Flowcharting Worksheet

Programmer: W. M. MAIN Program No.: Date: Page: 1 of 7
Chart ID: Chart Name: Program Name: DESIGN



IBM Flowcharting Worksheet

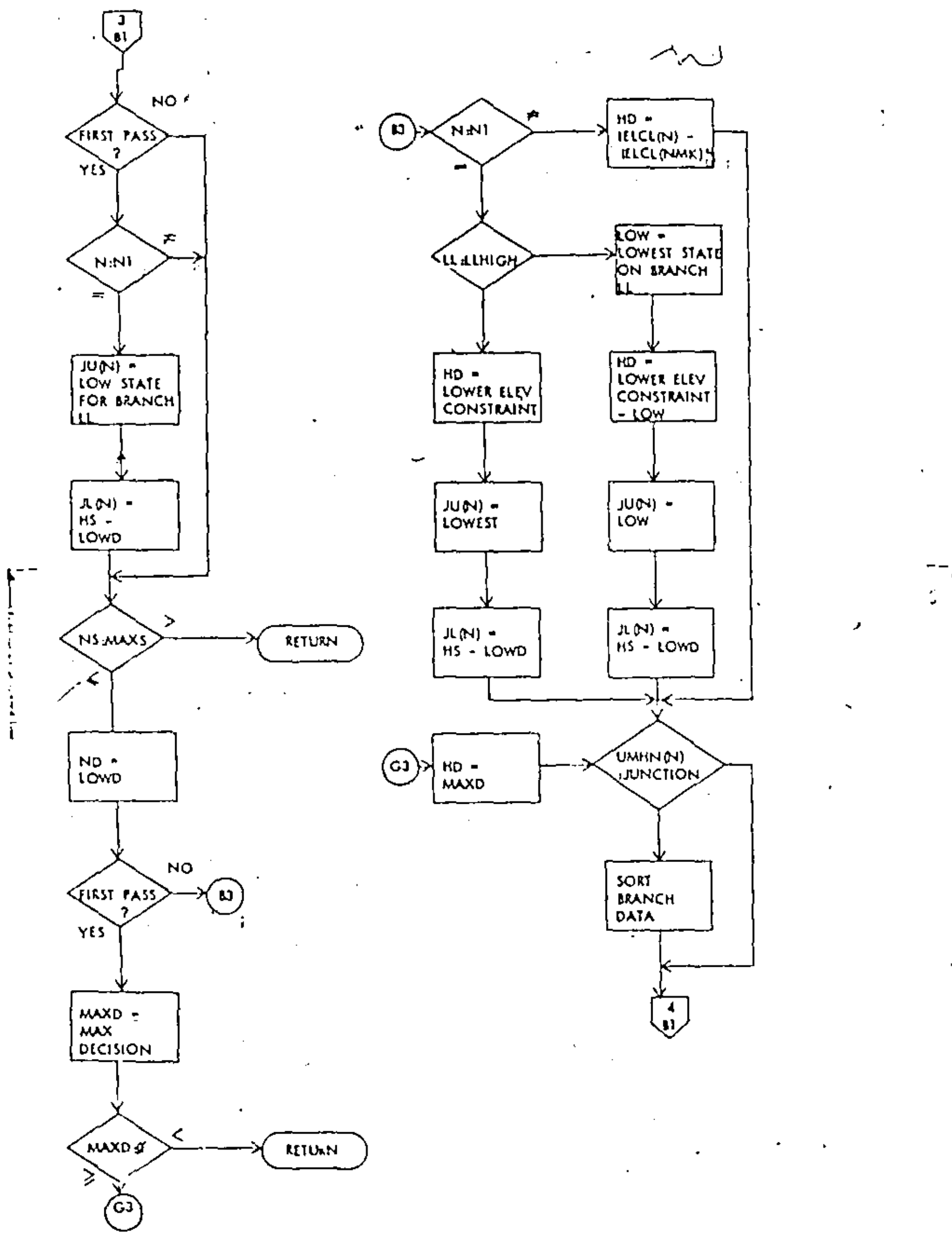
Programmer: W. M. MAIN Program No.: Date: Page: 2 of 7
Chart ID. Chart Name: Program Name: DESIGN



IBM Flowcharting Worksheet

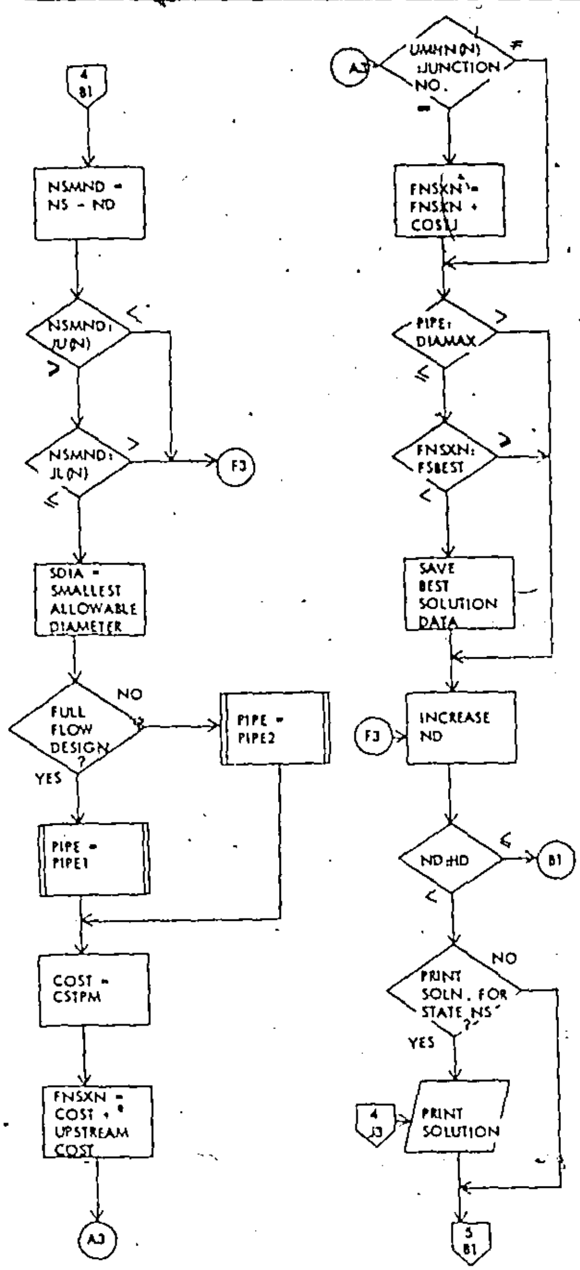
Programmer: W. M. MAIN Program No.: Date: Page: 3 of 7

Chart ID: Chart Name: Program Name: DESIGN



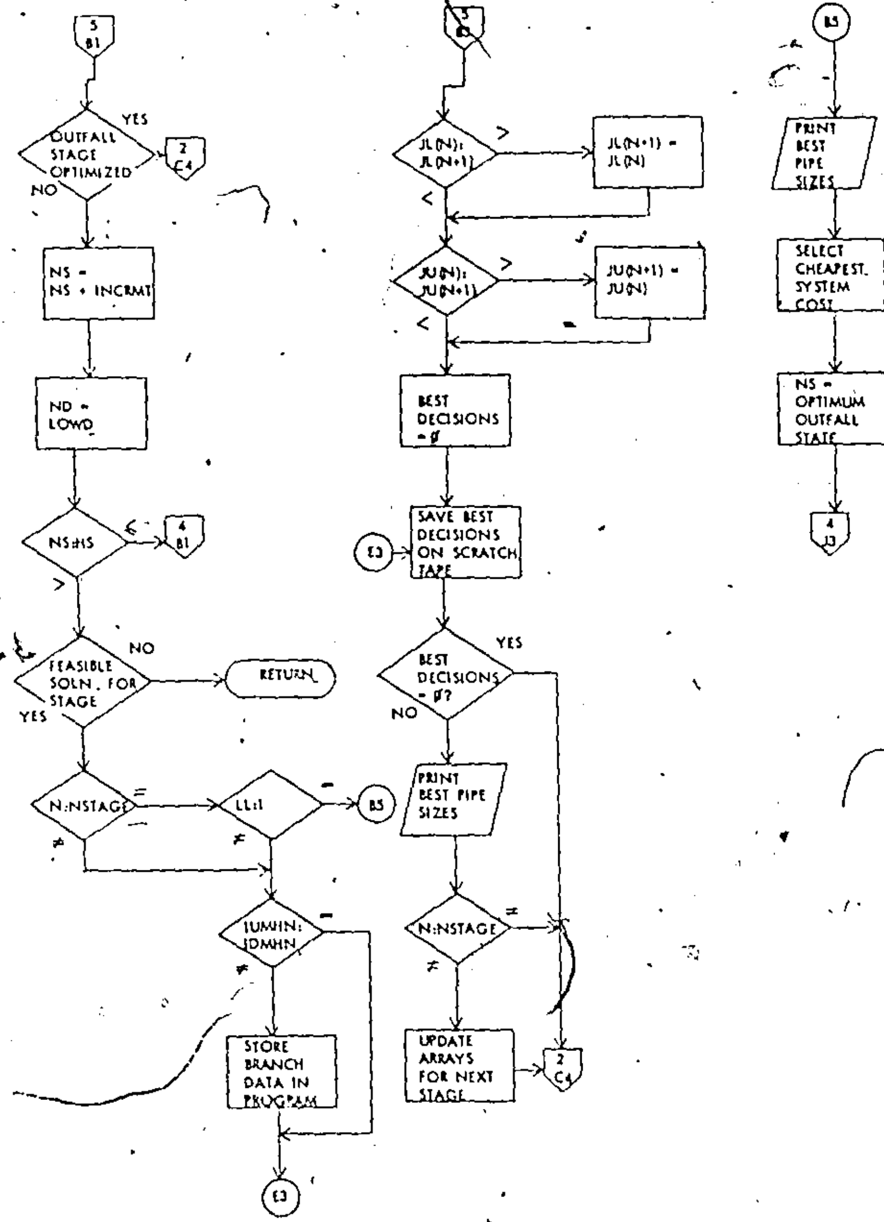
IBM Flowcharting Worksheet

Programmer: W. M. MAIN Program No.: Date: Page: 4 of 7
Chart ID: Chart Name: Program Name: DESIGN



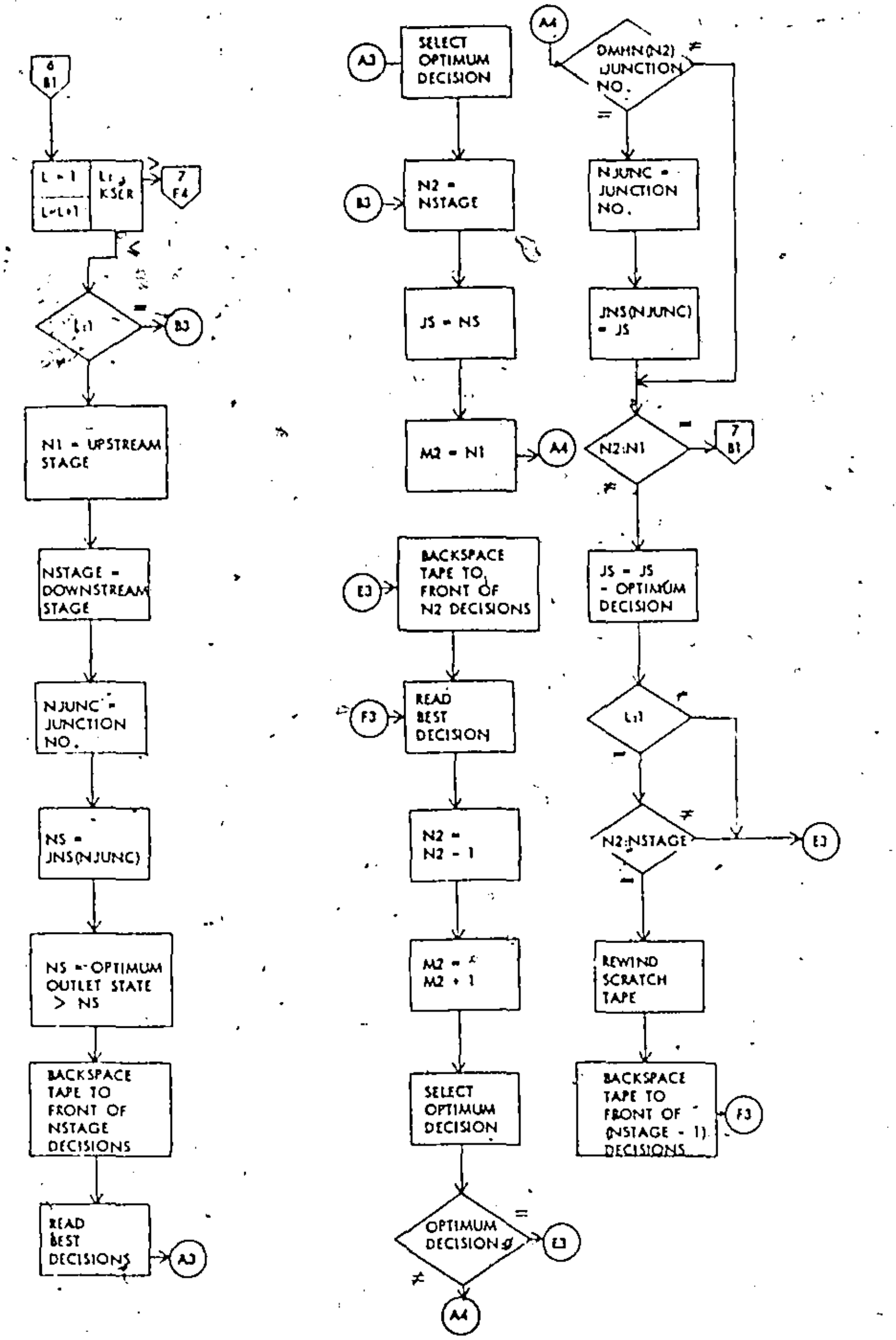
IBM Flowcharting Worksheet

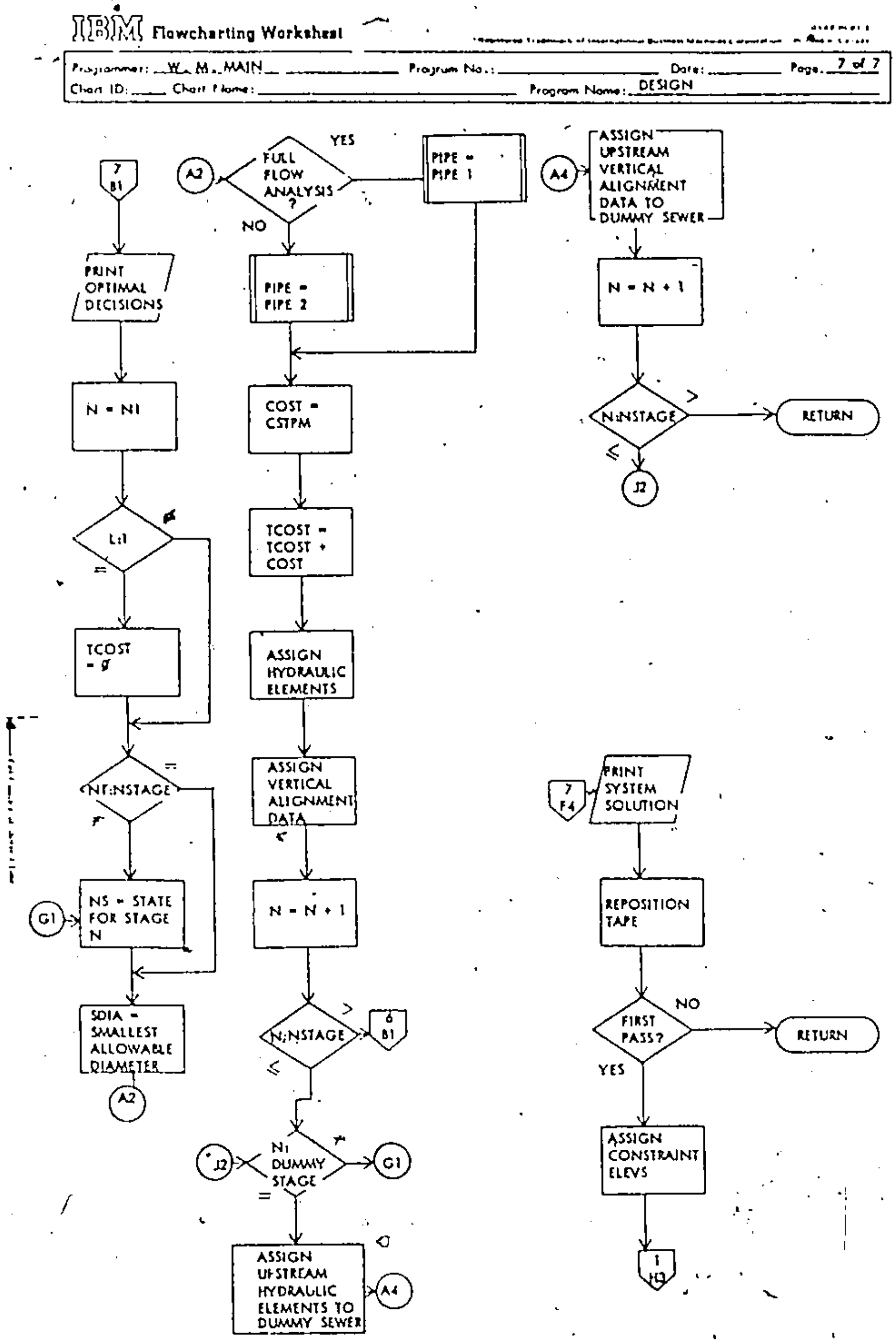
Programmer: W. M. MAIN Program No.: Date: Page: 5 of 7
Chart ID: Chart Name: Program Name: DESIGN

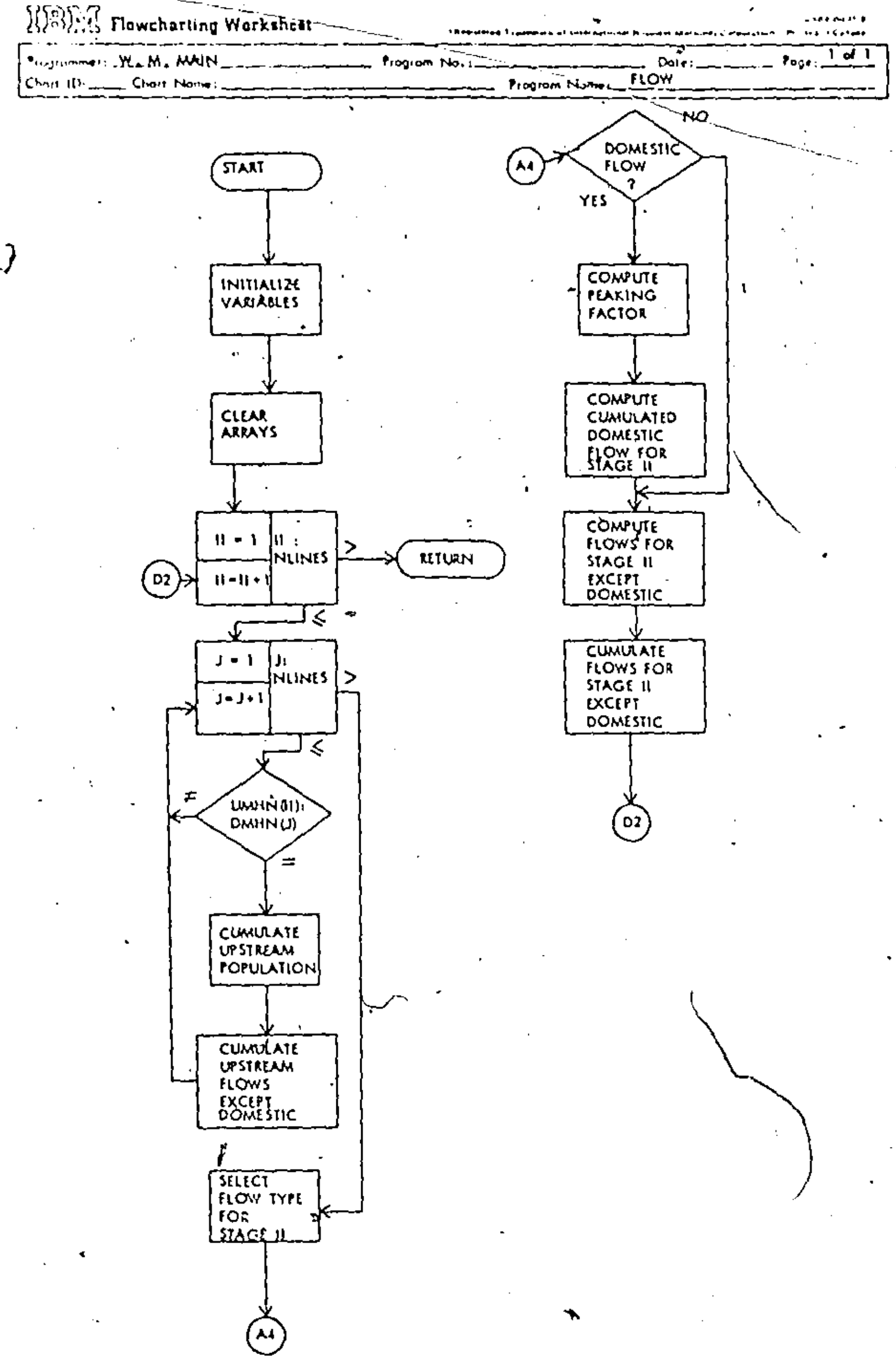


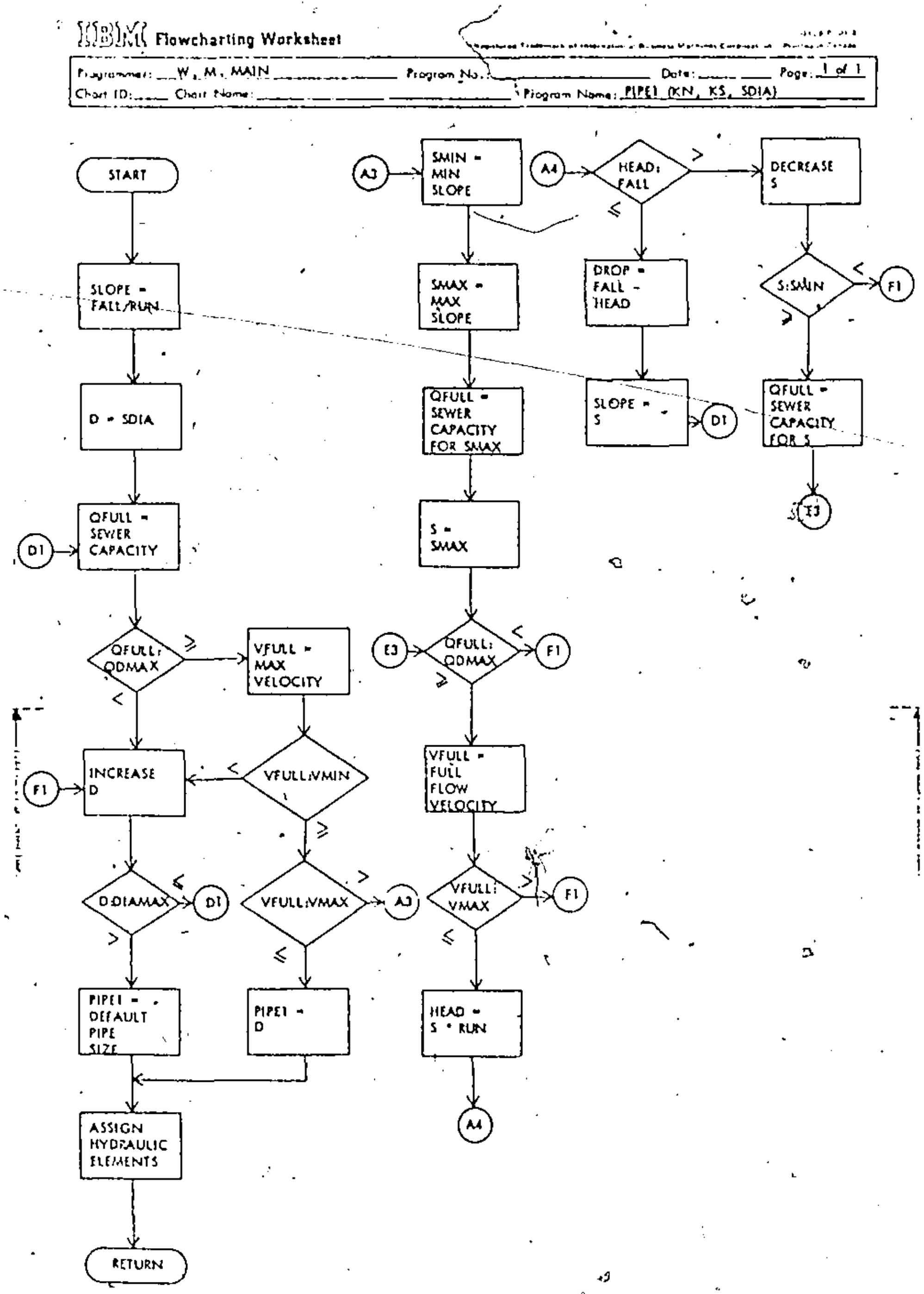
IBM Flowcharting Worksheet

Programmer: W. M. MAIN Program No.: Date: Page: 6 of 7
Chart ID: Chart Name: Program Name: DESIGN



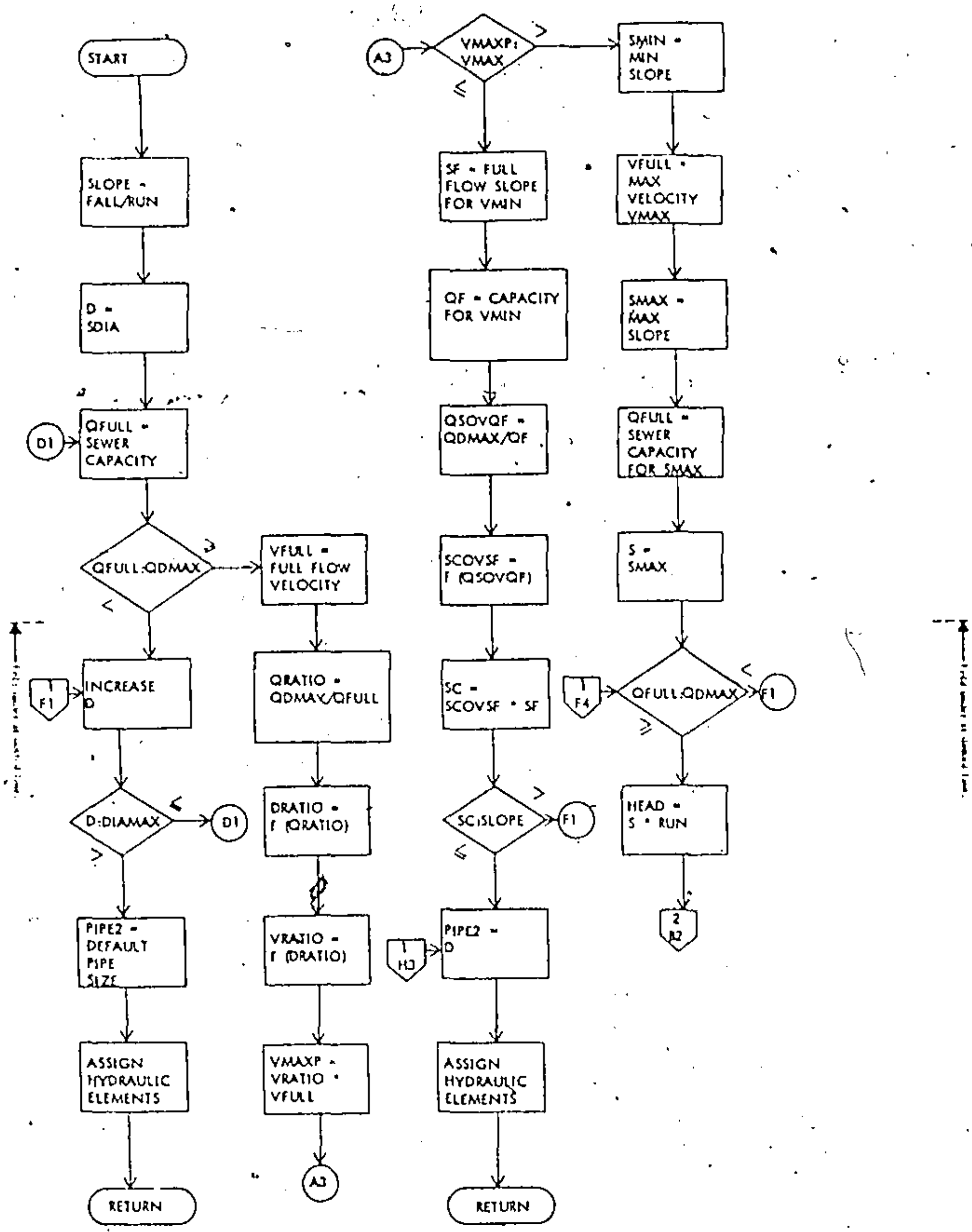

IBM Flowcharting Worksheet
Programmer: W. M. MAIN
Program No.:
Date:
Page 7 of 7
Chart ID:
Chart Name:
Program Name: DESIGN
7 B1
PRINT OPTIMAL DECISIONS
N = N1
L:1
TCOST = Ø
N1:NSTAGE
G1
NS = STATE FOR STAGE N
SDIA = SMALLEST ALLOWABLE DIAMETER
A2
FULL FLOW ANALYSIS ?
YES
NO
PIPE = PIPE 1
PIPE = PIPE 2
COST = CSTPM
TCOST = TCOST + COST
ASSIGN HYDRAULIC ELEMENTS
ASSIGN VERTICAL ALIGNMENT DATA
N = N + 1
N:NSTAGE
6 B1
J2
N: DUMMY STAGE
G1
ASSIGN UPSTREAM HYDRAULIC ELEMENTS TO DUMMY SEWER
A4
ASSIGN UPSTREAM VERTICAL ALIGNMENT DATA TO DUMMY SEWER
N = N + 1
N:NSTAGE
RETURN
J2
7 F4
PRINT SYSTEM SOLUTION
REPOSITION TAPE
FIRST PASS?
NO
RETURN
YES
ASSIGN CONSTRAINT ELEVS
1 H3


IBM Flowcharting Worksheet
Programmer: W. M. MAIN
Program No.:
Date:
Page: 1 of 1
Chart ID:
Chart Name:
Program Name: FLOW
START
INITIALIZE VARIABLES
CLEAR ARRAYS
II = 1
II = II + 1
II : NLINES
RETURN
D2
J = 1
J = J + 1
J : NLINES
UMHN (II) : DMHN (J)
CUMULATE UPSTREAM POPULATION
CUMULATE UPSTREAM FLOWS EXCEPT DOMESTIC
SELECT FLOW TYPE FOR STAGE II
A4
DOMESTIC FLOW ?
YES
NO
COMPUTE PEAKING FACTOR
COMPUTE CUMULATED DOMESTIC FLOW FOR STAGE II
COMPUTE FLOWS FOR STAGE II EXCEPT DOMESTIC
CUMULATE FLOWS FOR STAGE II EXCEPT DOMESTIC
D2


IBM Flowcharting Worksheet
Programmer: W. M. MAIN
Program No.:
Date:
Page: 1 of 1
Chart ID:
Chart Name:
Program Name: PIPE1 (KN, KS, SDIA)
START
SLOPE = FALL/RUN
D = SDIA
QFULL = SEWER CAPACITY
D1
QFULL: QDMAX
≥
<
VFULL = MAX VELOCITY
INCREASE D
F1
VFULL:VMIN
D:DIAMAX
VFULL:VMAX
A3
PIPE1 = DEFAULT PIPE SIZE
PIPE1 = D
ASSIGN HYDRAULIC ELEMENTS
RETURN
SMIN = MIN SLOPE
SMAX = MAX SLOPE
QFULL = SEWER CAPACITY FOR SMAX
S = SMAX
E3
VFULL = FULL FLOW VELOCITY
VFULL: VMAX
HEAD = S * RUN
A4
HEAD: FALL
DROP = FALL - HEAD
SLOPE = S
DECREASE S
S:SMIN
QFULL = SEWER CAPACITY FOR S
E3

IBM Flowcharting Worksheet

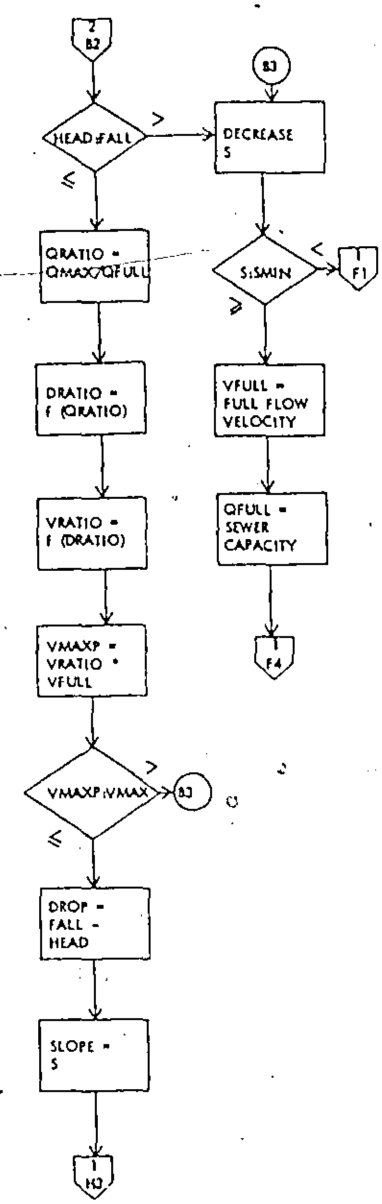
Programmer: W. M. MAIN   Program No.:   Date: Page: 1 of 2
Chart ID: Chart Name: Program Name: PIPE2 (KN, KS, SDIA)



IBM Flowcharting Worksheet

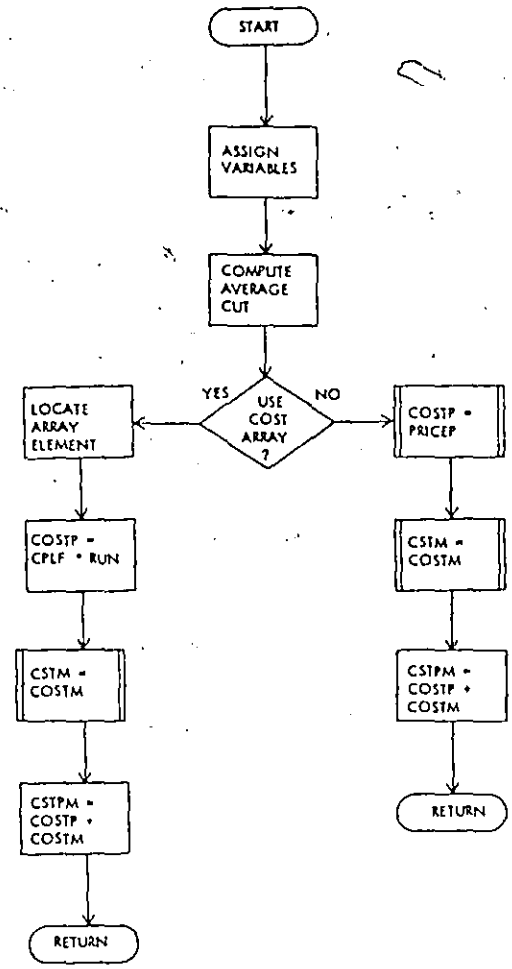
Programmer: W. M. MAIN Program No.: Date: Page: 2 of 2

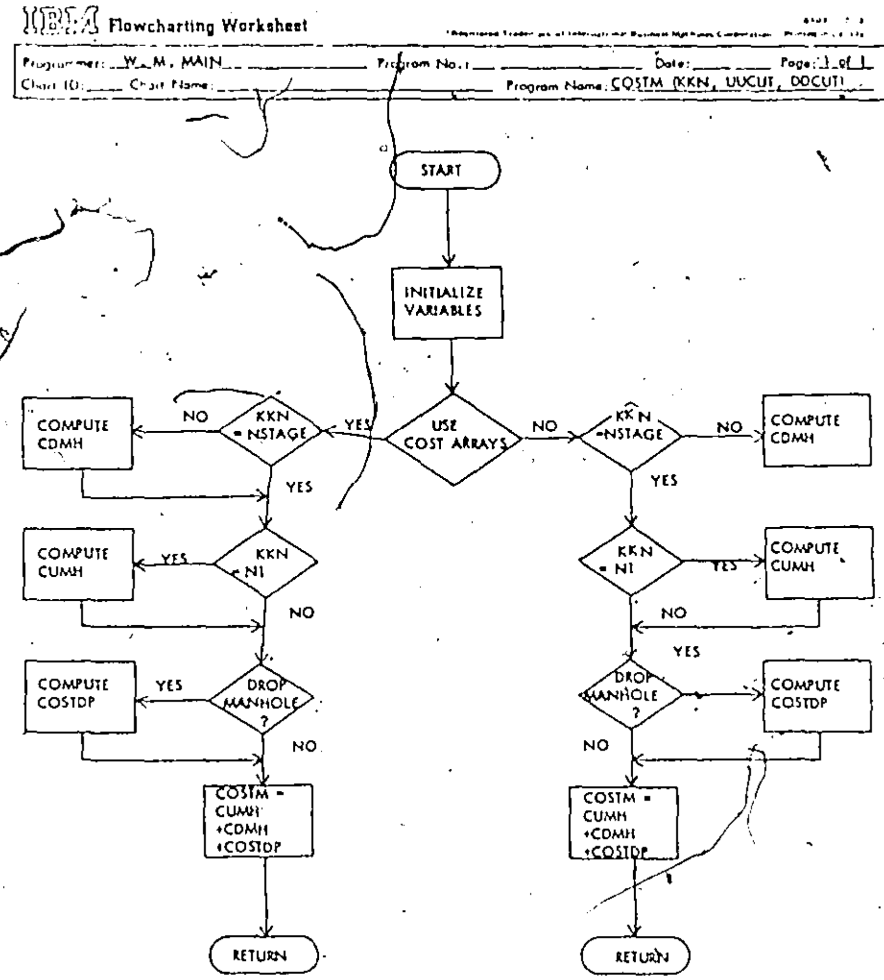
Chart ID: Chart Name: Program Name: PIPE2 (KN, KS, KX)



IBM Flowcharting Worksheet

Programmer: W. M. MAIN Program No.: Date: Page: 1 of 1
Chart ID: Chart Name: Program Name: CSTPM (KN, KS, KC)





IBM Flowcharting Worksheet
Programmer: W. M. MAIN
Program No.:
Date:
Page: 1 of 1
Chart ID:
Chart Name:
Program Name: COSTM (KKN, UUCUT, DDCUT)
START
INITIALIZE VARIABLES
USE COST ARRAYS
YES
NO
KKN = NSTAGE
COMPUTE CDMH
KKN = N1
COMPUTE CUMH
DROP MANHOLE ?
COMPUTE COSTDP
COSTM = CUMH +CDMH +COSTDP
RETURN