MODELLING OF REGIONAL SANITARY SEWERAGE NETWORKS FOR MINIMUM COST

by

© DAVID EDWIN MAUNDER, B.Sc. Engr.

A Thesis
Submitted to the School of Graduate Studies
in Partial Fulfillment of the Requirements
for the Degree
Master of Engineering

McMaster University
July 1980

MODELLING OF REGIONAL SANITARY SEWERAGE NETWORKS FOR MINIMUM COST

MASTER OF ENGINEERING (1980)
Civil Engineering

McMASTER UNIVERSITY
Hamilton, Ontario

TITLE: MODELLING OF REGIONAL SANITARY SEWERAGE NETWORKS FOR MINIMUM COST

AUTHOR: David E. Maunder, B.Sc. Engr. (McMaster University)

SUPERVISOR: Dr. Alan A. Smith

NUMBER OF PAGES: xii, 238

ABSTRACT

A computer model is developed for the solution of regional sanitary sewerage systems on a minimum cost basis. The regional system is considered to consist of a substantial number of alternative transportation (collection) routes, several potential treatment facilities and a receiving body of water which is able to accept the effluents from these facilities. The model, using a steady state analysis, facilitates the selection of a good collection network from the large number of possible networks, the selection of good depth-diameter combinations for each link in this final network and the selection of the size and number of treatment facilities which should be involved in the regional system. In addition, the required treatment efficiency of each facility will be determined so that a specified water quality goal may be satisfied.

A major point to be considered in attempting to select a least cost system is the tradeoff in costs between constructing a few, large, centralized treatment facilities (thereby benefitting from the economies of scale which generally exist) and the extra cost associated with transporting the wastes an extra distance to these centralized plants. The possibility of higher treatment

efficiencies being required at these centralized treatment facilities (due to a greater stress being exerted on the water quality of the receiving body of water) should also be considered. The piping costs are a function of the wasteflow. Fixed costs may be included in the cost function. The treatment plant costs are a function of both wasteflow and treatment efficiency. Fixed costs may also be included when determining the treatment plant costs.

Linear programming and dynamic programming techniques are used as a basis for the computer model. Additional subroutines were required to ensure that the technical constraints (for example, the minimum and maximum velocity constraints) were satisfied and to provide the missing components which are necessary to develop a complete computer package.

ACKNOWLEDGEMENTS

I would like to thank the National Research Council of Canada for the financial support made available to me during the course of this study.

A special appreciation goes out to my supervisor, Dr. Alan A. Smith who provided me with invaluable guidance throughout this study.

Thanks are also due to Janet Delsey of the Word Processing Center for typing the thesis.

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

TABLE OF CONTENTS (continued)

LIST OF FIGURES

LIST OF FIGURES (continued)

LIST OF FIGURES (continued)

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Introduction

In the past it has been common for each community producing liquid waste to be served by a local wastewater treatment facility.

In recent years with the growth of regional government and rapidly expanding urbanization, the realization of possible cost savings and the increasing concern for our rivers, lakes and streams has promoted the idea of a number of communities within a geographical region being served by a single (or very few) centralized treatment plants.

One primary reason for this trend is the potential saving in cost due to the economies of scale associated with the building and operation of large wastewater treatment facilities.

A study performed for the United States Environmental Protection Agency in 1976 [12] showed for example, that a 1.0 million gallon per day (mgd) activated sludge plant cost $2,200,000, whereas a 5.0 mgd plant costs about $1,600,000 per mgd of capacity.

However, there are two additional costs which may offset these savings. One is the extra cost of transporting the liquid waste to these larger facilities. The other extra cost is due to the possibility of having to provide higher levels of treatment at the larger facilities. This may be required because with centralization more waste is concentrated at a given point in the river and a greater stress is imposed on water quality.

The idea of regional planning has led to the development of mathematical decision models to aid the engineer or planner in selecting the proper sanitary sewerage system from the vast number of available alternatives.

To find the best sanitary sewerage system for a geographical area with several communities and/or industries the planner must answer the following questions:

1) Which communities should be included in the regional network?
2) What is the optimal system layout which will provide transportation of the liquid wastes to the treatment facilities?
3) What is the depth-diameter combination for each link in this layout?

4) From a number of possible wastewater treatment facilities, what are the sizes, treatment efficiencies and location of each plant which will, in conjunction with the transporation network provide a least cost and satisfy the water quality standards for the receiving body of water?

It should be pointed out that any mathematical model must make several physical and economic simplifications. As such the model should be viewed as a decision making aid or screening device to be used in the planning process. Also, mathematical models tend to find solutions that consider economics only. Due to aesthetic or political reasons the least cost economic solution may not be the best overall solution. Thus, the model should be flexible allowing the user to compare alternative plans to the least cost economic solution.

1.2 Simulation of the Physical Network

Figure 1.1 illustrates a hypothetical regional sewerage network. Within any network there are a number of possible locations for treatment facilities. The number and location of these plants will be dictated by topography, land availability and the presence of a receiving body of water which is able to accept the effluent from the treatment plants.

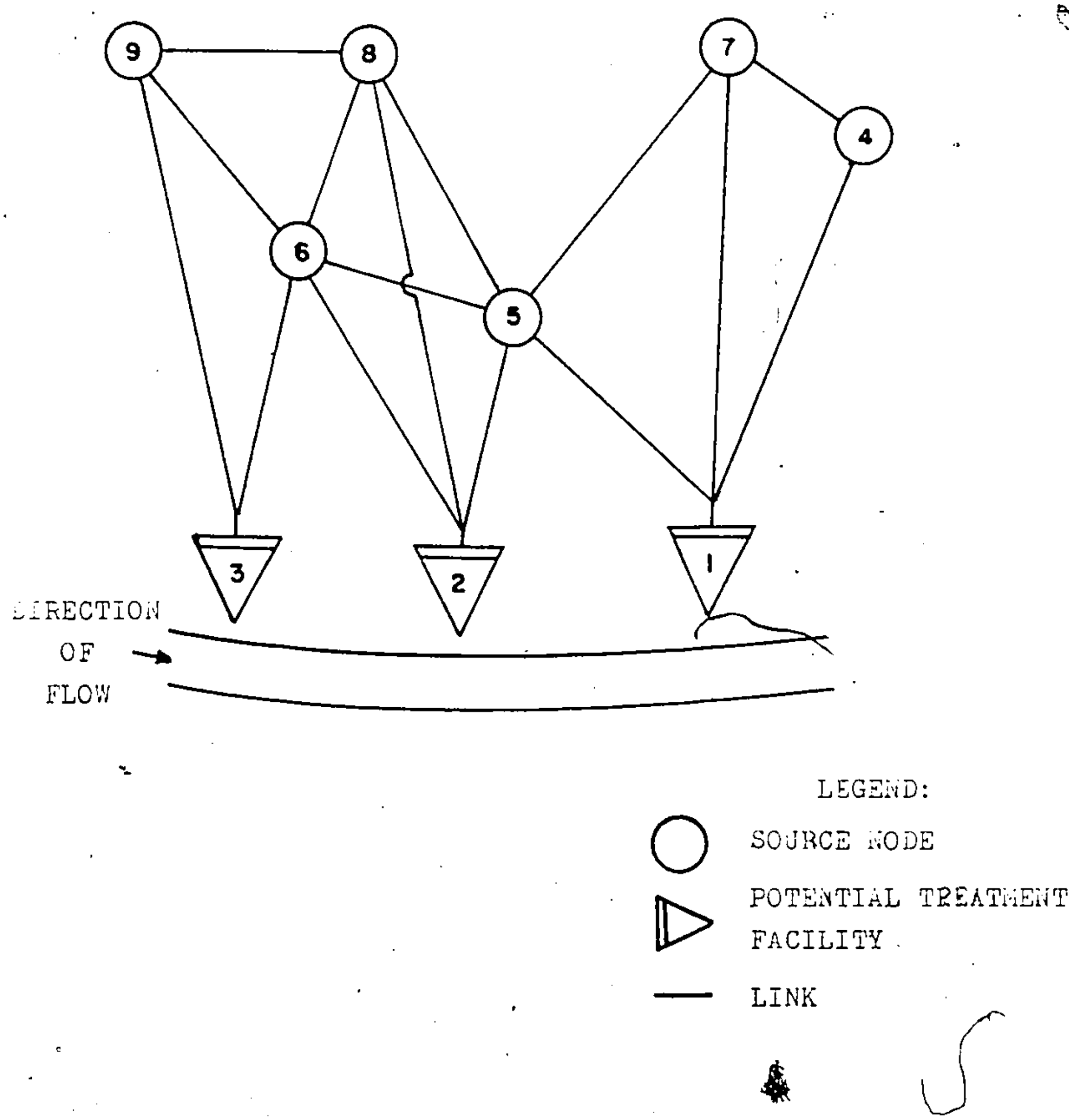


FIGURE 1-1 - TYPICAL REGIONAL SEWERAGE NETWORK

The geographical area under consideration is divided into sections. The supply of wastewater from each section is assumed to be concentrated at a node and it is assumed that all of the liquid waste flow originates from that node. How accurately the physical system is represented by this model will depend on the number of nodes chosen to represent the area. In some cases it may be advantageous to subdivide a community into a number of supply nodes to conform to the geographical layout of the region. Also, several nodes may be required to represent one community if differential growth and thus flowrates are to occur over the design period.

The topography of the region will determine the ground elevation at each supply node and treatment facility. The transportation of the liquid waste from the supply nodes to the treatment plants is facilitated by a series of links. The number of links, or alternative routes included in the initial problem will be decided by the engineer, and will in general depend on whether or not a link is considered to be economically and technologically feasible. The quantity of flow from each node is the summation of the commercial, industrial and residential flows as well as infiltration.

The abstraction of the physical system in the form of a linear graph or network allows the problem to be considered as an exercise in mathematical programming. In the chapters which follow the characteristics of each of the four subproblems (Chapter 1, section 1) are considered in detail.

Several different optimization techniques were considered during the course of this study. Two of these optimization techniques are used as a basis to develop a computer program package. The final computer package, when combined with engineering judgement allows preliminary planning of a sewerage network together with the location, sizing and levels of treatment for the processing centres. Although not without certain shortcomings which may be improved or refined in future research, the package is a practical solution to the problem of planning least cost regional sanitary sewerage networks.

CHAPTER 2

LITERATURE REVIEW AND STATEMENT OF OBJECTIVES

2.1 The Transportation Network

A large number of mathematical decision models have been developed by many authors in the last fifteen years. The earlier models commonly attempted to incorporate only one of the problems involved in the regionalization process. With the increased knowledge authors, in recent years, have tackled several aspects of the regionalization problem.

The selection of the best transportation network for the liquid waste can be broken down into two separate problems. The selection of the best route or system layout from a number of alternative routes must be made. In addition, the best depth-diameter combination for each link in this route should be found.

Dajani, et al. [8] attempted to answer the latter question. The problem was formulated initially in a nonlinear programming format and then manipulated into a linear programming problem by piecewise linearization. The piecewise transformation of the nonlinear objective function permits the solution to be obtained by a linear programming algorithm. The goal is to minimize the objective function subject to six technical and six linear approximation constraints for each link in the network. The solution yields the optimal values of the sum and the difference of the invert elevations of each link. Continuous diameters for each link can then be found.

The program has limited practical use due to the large execution time required and because of excessive storage requirements. Each link in the network requires sixteen rows in the problem matrix. The solution for a network consisting of 20 links was found in 60 seconds on a CDC-6400.

Dajani and Hasit [9] coupled integer programming with the convex-separable formulation of Dajani, et al. to arrive at discrete diameter solutions. The problem of the solution being in terms of continuous diameters was thus eliminated. The objective function can now be minimized subject to six technical constraints, six linear constraints, and two mixed integer constraints for each link in the network.

The addition of the two extra constraints increases the already large number of variables in the problem. In addition, the execution time is significantly increased. During the course of this research it was found that discrete pipe sizes could be found using linear programming without the addition of any mixed integer constraints (see Appendix B). Therefore, the extra variables and associated execution time could be avoided.

Gupta, et al. [14] applied nonlinear programming to the design of minimum cost sewerage networks. Powell's method of conjugate directions was employed to find the least cost design. Nonlinear programming algorithms do not require that approximations similar to those required in linear programming be used and therefore, may well provide a more accurate final solution. However, an efficient nonlinear programming algorithim has not yet been developed.

A six-link network was solved by Gupta, et al. The execution time required was 70 seconds on an IBM-360/44 computer. Compilation time required an additional 75 seconds. As such, this nonlinear program would be of no value in solving networks of any significant size.

Argaman, Shamir & Spivak [3] developed a dynamic programming solution for the simultaneous optimization of the layout and the depth-diameter combinations for the links in the layout. The method is very restrictive, however, since an important assumption of the model is that the direction of flow in every pipe is fixed. The authors suggest that in cases where the direction of flow is uncertain that the optimization be performed in both directions. The obvious implication is that if many alternative routes are available then the search for the optimal solution will become exhaustive. Indeed a 15 node network with a reasonable number of alternative system layouts required 25 minutes of computer time on an IBM 360/50 computer before the best solution was found.

The papers discussed previously show clearly that none of these mathematical models could efficiently find, from a number of possible system layouts, the best layout and the corresponding optimal depth-diameter combinations for each link in the layout. For this reason it was decided, in the present study, to approach this problem in a different manner. Whereas Dajani, et al. used linear programming to find the best depth-diameter combination for each link in a specified layout this author used linear programming primarily for finding a good layout among the many alternatives, and secondly as an aid in finding a good depth-diameter combination for each link in this layout. That the resultant mathematical model does find both a good layout and depth-diameter combinations will be shown by a comparison with a dynamic programming model in a later chapter.

2.2 Centralization of Treatment Facilities

The papers which are reviewed subsequently discuss the problem of centralization of the treatment facilities, that is, of finding the number and location of the treatment facilities and the assignment of the liquid wastes to the plants such that the total cost is minimized.

Brill and Nakamura [6] used a branch and bound method for use in evaluating alternative regional wastewater treatment systems. A set of linear and nonlinear constraints describe the physical constraints and the large number of alternative locations and sizes of plants and interceptors. Feasible solutions are then generated and examined in sequence. Each feasible solution specifies a configuration of treatment plants and interceptors. The procedure can be followed until the least costly configuration has been found.

The objective function is described in terms of the treatment plant and interceptor costs. Both of these costs are functions of flow and are classified as concave functions (i.e. exhibit economies of scale with respect to flow, see Figure 3.2 and Figure 3.3).

Brill and Nakamura have noted that any branch and bound method tends to become impractical for problems with many concave functions. The authors, therefore, have had to greatly simplify the number of potential transportation routes so that the problem can be solved within a reasonable amount of computer time. However, overly simplifying the problem to one where there are only one or two possible transportation routes to each potential treatment plant means that the best solution found by the branch and bound method may well not be a good solution. This is due to the fact that the simplification of the transportation network does not allow the user to accurately define the tradeoff in cost between using a few centralized treatment plants and the additional costs that are required to transport the wastes the extra distance to these centralized plants. As has been stated previously, this is one of the major problems which must be solved if a good model is to be developed.

One of the assumptions made by Brill and Nakamura was that fixed treatment requirements are assumed. Thus, treatment costs are a function of flow only. To completely address the problem of regional optimization the questions of where the treatment facility's effluent is to be released and the effects of releasing these effluents should be addressed.

Typically, these wastes are discharged into a receiving body of water such as a river, estuary, or lake. These bodies of water generally have an upper limit as to the amount of waste they can accept. If this limit is exceeded the waters may become anaerobic, and will be turbid or dark. In addition, all fish life will be jeopardized.

The assumption of fixed treatment efficiencies at each plant totally disregards the effects of the receiving body of water and is harmful for economic and non-economic reasons. If the assumed fixed treatment efficiency is too low then the receiving body may well become anaerobic or toxic to fish life as previously described. If the level of treatment is too high this will result in treatment efficiencies that are unwarranted and economically expensive.

Jarvis, et al. [16] also developed a computer program to design regional wastewater systems. The authors assumed that the quantity of wastewater entering the system was proportional to the number of population units involved. Increasing economies of scale were assumed to be present and both transportation costs and treatment costs are concave functions of population capacity. The authors thus considered it appropriate to approximate the cost functions with sets of linear segments that could be managed with a fixed charge algorithm.

The algorithm used allows the user to accurately define the transporation network, thus eliminating a shortcoming which occurred in Brill and Nakamura's method. The authors also acknowledge the fact that the development of regional wastewater treatment system plans requires that factors such as environmental impact of the discharged wastes be considered. The objective function used, however, indicates that the authors considered plant capacity to be the only significant variable used in determining plant cost. Water quality considerations were therefore not accounted for.

The major cost of a sewer link is the excavation and backfill costs. Walsh and Brown [34] stated that excavation and backfill costs may account for as much as 80 percent of the total cost of the sewer. This percentage, will in general, increase as the excavation depth increases.

To accurately define the link costs the total cost should be a function of the pipe diameter used as well as the excavation and backfill costs. Jarvis, et al. have simplified this cost to be dependent only on the pipe capacity. This presumes a correlation between depth and diameter which in fact can only be relied upon in general terms and not for each specific case.

The assumption that all pipes will be laid at the minimum allowable depth may well be true in areas where the terrain is steep; it is not however, likely to be valid in areas which have gently sloping terrain. Thus, the representation of the link cost as a function of diameter only will, in areas with gently sloping terrain lead to a cost which does not accurately represent the true cost for the link.

In addition, as stated in Chapter 6, section 5, the invert elevation of a downstream pipe must be as low as or lower than the invert level of all pipes directly upstream to the pipe. This states that there is an interdependence between the links and that the cost of a downstream link will be higher due to extra excavation and backfill costs if the incoming elevation of an upstream pipe is low. By ignoring the excess excavation and backfill costs Jarvis, et al. have ignored the interdependence between pipe elevations. Ignoring this interdependence will also lead to costs which do not accurately represent the true link costs.

Chiang and Lauria [7] approached the problem of regionalization in a slightly different manner. A heuristic algorithm which allows for construction at different stages within a finite planning period is used in an attempt to optimize the regional system.

The user initializes the program by proposing a configuration consisting of treatment facilities and transportation routes which he feels are the best for the region in each time period. The proposed plan is then evaluated to explore the possibility of combining two or more spatially distinct plants (spatial comparison) followed by the comparison to determine whether it is less expensive to combine two or more temporally distinct plants and transportation routes (temporal comparison). If a cost reduction is found then the best solution is updated.

The incorporation of multi-construction periods into the algorithm may well lead to final costs that are lower than those that would be found if only the beginning and end of the design period was considered (a two stage problem).

However, the computer storage and computation time required are directly proportional to the number of construction stages used.

Chiang and Lauria did not encounter excessive execution time or storage problems when solving several example systems. For a problem with 20 nodes and 5 construction opportunities the computation time was 72 seconds, and the storage requirement was 190 K on an IBM 370/165.

Their representation of the physical network was however, simplified in a similar fashion to the methods used by Brill and Nakamura and Jarvis et al. In addition to these simplifications, if this algorithm is to be efficient the initial network chosen by the user should contain at most one or two circuited networks (see Chapter 3, section 1). Increasing the number of circuited networks greatly increases the number of alternative transportation routes and thus the computation time required to find the best solution. This greatly restricts the application of this method since in larger networks the planner may find it difficult to reduce the number of alternative transportation routes and at the same time ensure that the final solution found is indeed a good solution.

2.3 Consideration of a Water Quality Standard

The consequences of assuming fixed treatment levels and thus ignoring the effects of the effluent from the treatment plants on the receiving body of water were touched upon earlier in this chapter. A more complete discussion will now be considered.

In recent years various protective standards have been proposed and implemented by regulatory agencies across North America in an attempt to contend with the increased contamination of the water courses with various kinds of materials. The stream standard places restrictions on the quality of water flowing in the watercourse and prohibits any user from discharging matter into the stream that depresses the water quality below a specified level. The effluent standard specifies the level of treatment the user must provide before discharge.

To answer the question of which one of these standards should be used in designing a regional system one should look at both economic and non-economic considerations. This is necessary since the knowledge of todays engineers and environmentalists does not allow us to measure solely in economic terms all the costs and benefits resulting from the development of a water resource. Several variables such as aesthetics, health, recreational aspects and value of fish life cannot be totally expressed in economic terms.

The use of a stream standard, commonly dissolved oxygen, allows us to at least obtain a partial understanding of the importance of some of these variables. For example, it is generally acknowledged that if fish life is to be prosperous then a dissolved oxygen concentration of at least 4 milligrams per litre (mg/l) be maintained. Aesthetically, the condition where the dissolved oxygen concentration is zero and putrification occurs is quite undesirable.

Therefore, if the uses of the watercourse are predetermined a water quality level can be set and the design can be completed with this standard in mind. On the other hand, the use of the effluent standard will not guarantee that a particular water quality standard can be met. Variables such as aesthetics and value of fish life would have to be completely ignored.

The economic advantage of using a stream standard as opposed to an effluent standard can be illustrated by reviewing a study performed by Liebman and Lynn [19] on the Williamette River in Oregon. A dynamic programming model was used to calculate the individual treatment plant efficiencies which would satisfy the specified minimum dissolved oxygen concentration at a minimum cost. These costs were then compared to costs found using the effluent standard criteria.

The Oregon State Standard Authority suggested that all treatment plants should provide at least 85 percent Biological Oxygen Demand (BOD) removal. Leibman dismissed this standard since the requirement was far more stringent than the dissolved oxygen standard that was set by the same authority. If an equal level of treatment (of BOD) at each plant was employed then the water quality standards could be satisfied if all plants operated at 55 percent efficiency. In addition, it was shown that if the individual plant efficiencies were allowed to vary that a further reduction in cost of 1.3 percent would occur.

ReVelle et al. [23] used linear programming to select the efficiencies of the treatment plants along a river while satisfying the dissolved oxygen standards. The objective function was structured only in terms of the treatment plant costs.

The procedure involved linearizing the cost vs. efficiency curve for each plant. The curve was assumed to be convex and was broken into linear segments. However, typical cost vs. efficiency curves as presented by Shih and DeFilippi [27] and Graves et al. [13] show that the cost curve is, in general, convex only for efficiencies above 75 to 80 percent (see Figure 2.1).

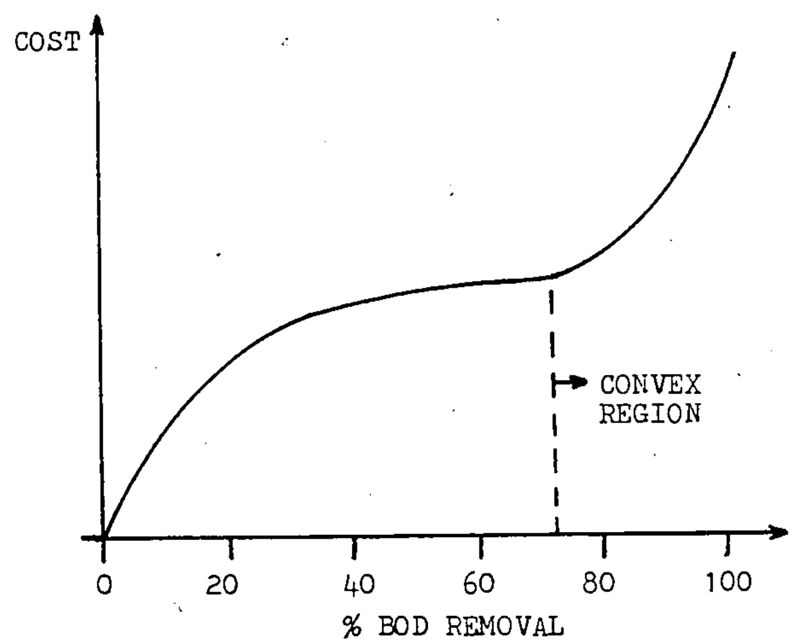


FIGURE 2·1 - TREATMENT PLANT COST vs. % BOD REMOVAL

In addition, if economies of scale are taken into account the cost vs. flow curve will be concave (see Figure 3.3). If the calculations of the efficiencies at each treatment plant are to be performed simultaneously, as in ReVelle, et al., then the efficiency chosen for each plant will depend on the relative flows at each plant, the incoming waste concentration at each plant and the water quality standards which have been set for the region. The result is that the cost curve is a function of both the efficiency and flow at each plant and will not exhibit the necessary properties (see Appendix C) to ensure that a good solution can be found using a linear programming algorithm.

In general the waste treatment cost function is nonlinear. Thus a nonlinear programming model would be a more adequate approximation to the real system than a linear model.

Bayer [5] presents a nonlinear program which may be used on branching river systems. The river is modelled as a closed system consisting of a main stream and several tributaries. The objective function along with the constraint set constitutes a nonlinear programming problem, the solution of which provides the set of wastewater treatment plant efficiencies necessary to achieve the water quality standard for the main stream.

Nonlinear models may indeed lead to more accurate answers than other optimization models. However, the difficulties in finding efficient algorithms for problems of any significant size greatly restricts the use of nonlinear programming.

Although no computation time is given by the author for the system solved within the paper, it may be safely assumed that the problem stated above would restrict the use of the author's nonlinear programming algorithm to small systems. Otherwise, the computational cost would be prohibitive.

O'Laoghaire [22] developed a model which studied the problem of a water quality management system undergoing a rapidly increasing environmental stress. The problem addressed by O'Laoghaire differs slightly from others in that there are several existing treatment plants along the river reach with fixed efficiencies. The objective is to determine the size and timing of construction of the new treatment plants in addition to finding an overall treatment plant operating policy (efficiencies for each of the new plants) so that environmental standards are maintained at a minimum cost. The computer model is a "zero-one" mixed integer programming model which is solved by decomposition into a branch and bound algorithm and a linear programming algorithm.

The model presented by O'Laoghaire is not likely to find a minimum cost for two reasons. A linear programming algorithm was used in an attempt to minimize the objective function (the objective function is stated in terms of the new treatment plants only). As was pointed out in the review of ReVelle's paper the solution found by linear programming, for this type of problem, is dependent on the shape of the cost curve. Thus, this method cannot guarantee that a least cost solution will be found.

Secondly, only a selected number of flow values can be used at each proposed treatment plant. Thus only a few combinations of treatment plant flows will be accounted for. Normally, if the transportation of wastes from one area to a plant further upstream or downstream is allowed then there would be an infinite set of flow combinations which could be possible. By restricting the number of combinations the full assimilative capacity of the receiving body of water to accept biodegradable wastes may not be fully accounted for. Also, by restricting the number of combinations of flows the tradeoff between economies of scale inherent in wastewater treatment plants and added pipe network collection costs cannot be properly dealt with. For these reasons a least cost solution is unlikely to be found.

Joeres et al. [17] developed a regional model to consider the tradeoff between the economies of scale inherent in wastewater treatment plants and the added piping costs. A network of allowable piping connections between sources and facilities is established and all cost functions are made piecewise linear with fixed costs. A mixed integer programming model is used to find the minimum cost pattern of treatment facilities and source assignments. The receiving bodies of water are several small lakes.

The method presented by the authors tackles the regionalization problem in a more thorough manner than the previous papers since a reasonable number of transporation routes are allowed and an attempt is made to satisfy the environmental standards.

Joeres et al., however, simplified the link costs such that the excavation depth was assumed to be constant. As was explained in the review of Jarvis et al., this will lead to costs which do not accurately represent the true cost for the link. In effect, only the best layout is being selected. The selection of good depth-diameter combinations is not being considered.

Joeres et al. also consider the environmental impact on the receiving bodies of water due to the discharging of wastes (BOD). The maximum loading allowable (pounds of BOD per day) is determined at each discharge point. The treatment efficiencies for all plants are assumed to be fixed at 85 percent. The maximum discharge capacities for each plant can then be determined. It is stated by Joeres et al. that this procedure will result in an overall minimum cost being found. In addition, the environmental standards which were imposed will be satisfied. However, if both the efficiencies and discharge capacity were allowed to vary at each plant (maximum loading would still be constant) then a lower overall cost would likely be found. This is due to the nonlinear aspects of the cost vs. flow and cost vs. efficiency curves.

The problem solved by Joeres et al. consisted of a 16 node problem with potential treatment plants being allowed at 12 of these nodes. The computation time required to find the minimum cost was 14.4 minutes on a Univac 1108. The rather lengthy computation time required to find the least cost solution combined with the large number of variables and constraints (192+136 respectively), would limit the use of this mixed integer program to problems of approximately this size.

Linear programming was found by this, and other authors to be a useful method in solving the transportation part of the regionalization problem. However, this method is not applicable for selecting the treatment plant efficiencies for variable flows, due to the uncertain shape of the cost curve. Other methods, such as nonlinear and mixed integer programming were shown to have major drawbacks.

Dynamic programming, on the other hand may be used with cost curves of any shape. In addition, dynamic programming is well suited for problems of this nature since the method will account for the interrelationship between the plants, that is, the efficiency selected at an upstream plant will effect the efficiency required at a downstream plant if a specific water quality standard is to be satisfied. Storage and execution time, the two major drawbacks of dynamic programming will not limit the use of the method since both storage and execution time are largely dependent on the number of stages (potential treatment facilities in this case). Normally the number of potential treatment facilities will be a reasonably small number (say 5 or 6).

The above literature review indicates that a mathematical model has not yet been developed which answers all of the questions which arise if the problem of regionalization of sanitary sewerage networks is to be properly addressed. It is this author's intention to address this problem in a more complete fashion than has previously been attempted.

2.4 Objectives of the Study

The objective of this research is to develop a computer model which will select, from a large number of possible transportation routes and several potential treatment facilities a good system layout, good depth-diameter combinations for each link in this layout and in conjunction with the selected layout the location and sizes of the treatment facilities. In addition, the required efficiencies of each plant selected will be determined such that a specified water quality goal may be satisfied.

CHAPTER 3

THE MATHEMATICAL MODEL

3.1 Constraint Equations

The sewerage network shown in Figure 1.1 may be described as a "circuited network". This type of network defines all possible routes and treatment plants which have been deemed economically and technologically feasible by the engineer. A circuited network, as the name implies, contains closed loops or circuits. In theory, if the network were non-planar, two links may cross without forming a node. In practice, for the system under study here it would not be practical to have links crossing without a junction.

For each link there are two flow variables, one in each direction. In any practical solution at least one of these variables will be equal to zero. The condition of non-negativity for the flow variables is thus implied.

The sign convention for all nodes (supply nodes and treatment centres) assumes that flow out of a node is positive and that flow into a node is negative.

For any supply node, the following condition must be satisfied:

The algebraic sum of all flows leaving a supply node must equal the required supply (stipulation) for that node.

Thus for node #9 of Figure 3.1, continuity is defined by the equation:

$$Q_1 - Q_2 + Q_5 - Q_6 + Q_7 - Q_8 = \text{stipulation} = S_9 \quad (3.1)$$

The right hand side is defined as the stipulation. It will be positive for supply nodes which implies an addition to the system.

For treatment facilities factors such as land restrictions, environmental constraints or existing plant capacities may provide an upper limit as to the amount of waste that can reliably be treated. Treatment plants abstract flow from the system and are therefore represented by stipulations (processing capacities) which are negative in sign.

The situation at any treatment facility is such that the algebraic sum of the inflows into the plant will be less than or equal to the processing capacity. The inequality can be changed to an equality by the addition of a slack variable. The slack variable represents the surplus processing capacity. Because of the sign convention adopted here, the sign of the slack variable is negative and simulates an additional flow into the treatment plant.

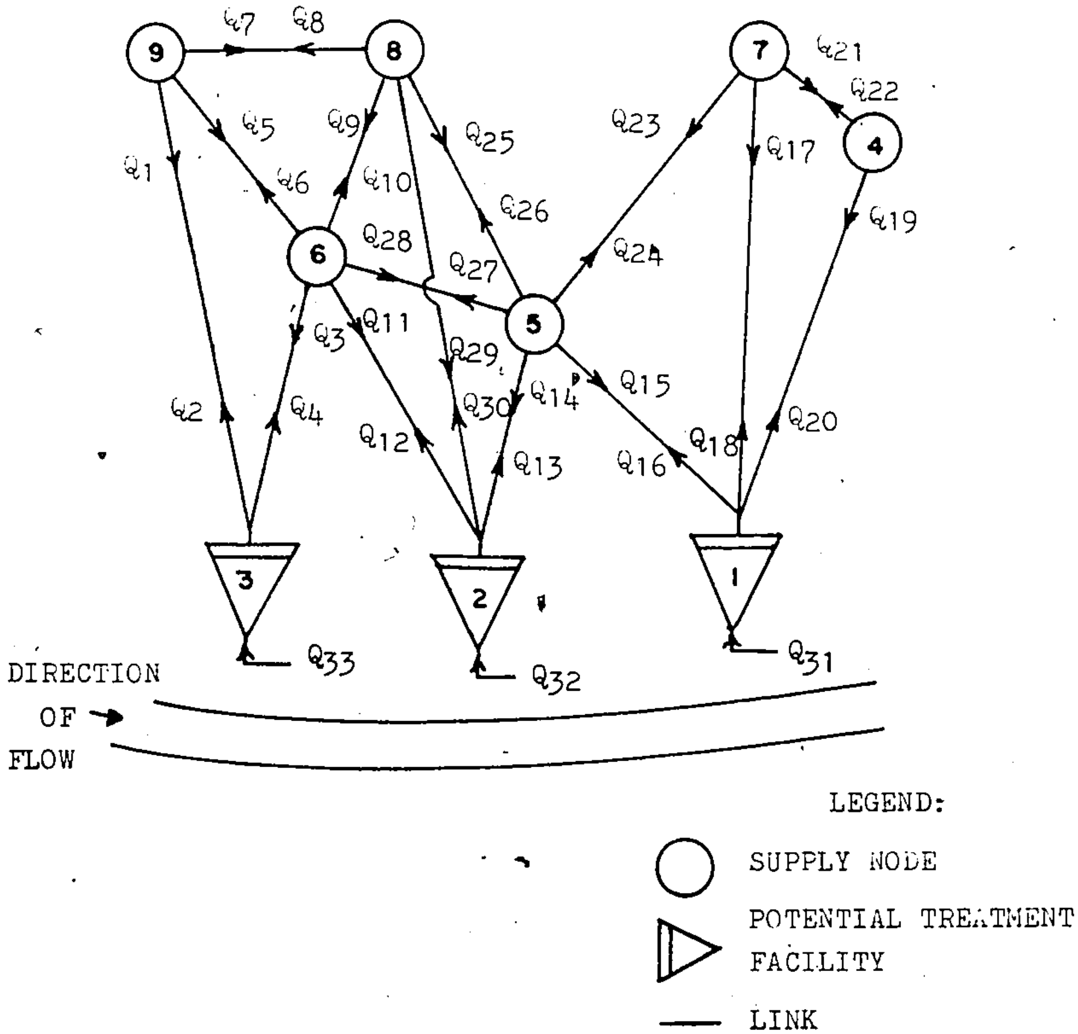


FIGURE 3·1 - GRAPHICAL REPRESENTATION OF A REGIONAL SEWERAGE NETWORK

Continuity at node #3, Figure 3.1 is then defined by the inequality:

$$-Q_1 + Q_2 - Q_3 + Q_4 \geq -\text{Stipulation} = -D_3 \qquad (3.2)$$

or, with the addition of the slack variable Q_{33} as

$$-Q_1 + Q_2 - Q_3 + Q_4 - Q_{33} = -D_3 \qquad (3.3)$$

or

$$Q_1 - Q_2 + Q_3 - Q_4 + Q_{33} = D_3 \qquad (3.4)$$

The equality constraints may be conveniently described by the matrix equation:

$$\overline{A} \times \overline{Q} = \overline{B} \qquad (3.5)$$

wh[illegible]e,

A = a matrix of structural coefficients which take the value of 0, 1, or -1 (N x NQ elements),

Q = a column vector of NQ flow variables,

B = a column vector of N constraint stipulations,

N = the number of nodes in the network for which constraints are written, and

NQ = the total number of flow plus slack variables.

The problem can be identified as a network problem by the fact that each column of the structural matrix has only two non zero entries (of value -1 and 1). Table 3.1 shows

NODE NO.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	STIPULATIONS
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	−1	+1	−1	+1	−1	+1	0	0	0	0	0	0	0	0	0	0	−1	0	0	−30
2	0	0	0	0	0	0	0	0	0	0	−1	+1	+1	−1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	−1	+1	0	−1	0	−30
3	−1	+1	−1	+1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	−1	−30
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	+1	−1	−1	+1	0	0	0	0	0	0	0	0	0	0	0	+2
5	0	0	0	0	0	0	0	0	0	0	0	0	−1	+1	+1	−1	0	0	0	0	0	0	−1	+1	−1	+1	+1	−1	0	0	0	0	0	+4
6	0	0	+1	−1	−1	+1	0	0	−1	+1	+1	−1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	−1	+1	0	0	0	0	0	+8
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	+1	−1	0	0	+1	−1	+1	−1	0	0	0	0	0	0	0	0	0	+6
8	0	0	0	0	0	0	−1	+1	+1	−1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	+1	−1	0	0	+1	−1	0	0	0	+3
9	+1	−1	0	0	+1	−1	+1	−1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	+5
FLOW VARIABLE ORIGIN-DESTINATION PAIRS																																		
U\S NODE #	9	3	6	3	9	6	9	8	8	6	6	2	2	5	5	1	7	1	4	1	7	4	7	5	8	5	5	6	8	2	0	0	0	
D\S NODE #	3	9	3	6	6	9	8	9	6	8	2	6	5	2	1	5	1	7	1	4	4	7	5	7	5	8	6	5	2	8	0	0	0	

TABLE 3·1 - STRUCTURAL COEFFICIENTS AND ORIGIN-DESTINATION PAIRS the structural coefficients together with a vector of stipulations for the configuration of Figure 3.1.

3.2 Objective Function

The costs for the collection and treatment of wastewater may be divided into two distinct parts, transportation and processing costs.

Transportation costs consist of costs due to the capital expenditure on sewers and pumps, the annual costs for operation and maintenance and sinking fund costs. Processing costs comprise capital expenditures on collection chambers, treatment works and ancillary equipment in addition to annual costs for treatment maintenance and sinking funds.

The transportation cost for one link may be stated in terms of the flow variable. The cost curve is concave, monotonic, nonlinear (demonstrating economies of scale) and may also exhibit a fixed charge implying a minimum charge limit. Figure 3.2 illustrates the transportation cost curve for one link. For a given flow there is one diameter and resulting average excavation depth which leads to a minimum cost. The smooth shape of the curve will occur only if an infinite number of diameters are available. In practice, and in this study, only a few discrete diameters are available. The effect on the cost curve as a result of limiting the number of available diameters is shown in

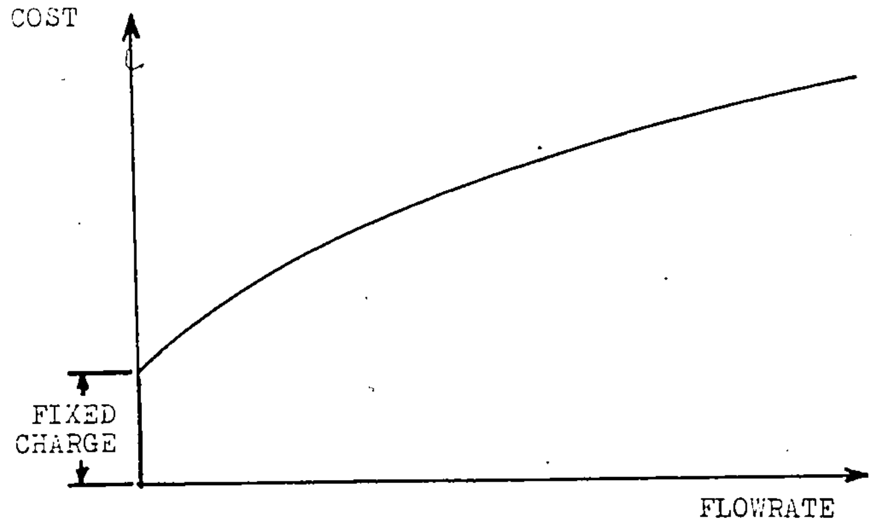


FIGURE 3·2 - TYPICAL TRANSPORTATION COST CURVE

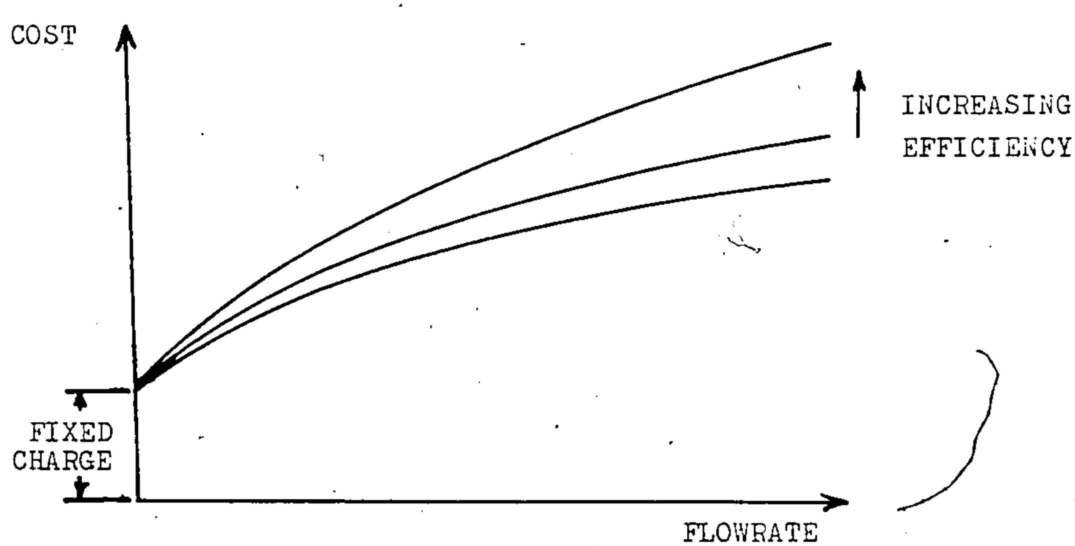


FIGURE 3·3 - TYPICAL PROCESSING COST CURVE

Appendix B. The fixed charge for the transportation cost curve is comprised of the smallest pipe set at the minimum allowable excavation depth.

The processing cost for a treatment facility may be stated in terms of the quantity of flow through the plant and the treatment efficiency of the facility. The curve of cost vs. flow, for a given efficiency is also concave, monotonic and nonlinear.

Since there are a number of possible operating efficiencies at each treatment plant there will be a number of these curves (see Figure 3.3). The fixed charge would reflect such items as land requirements and entrance road costs.

The curve of cost vs. treatment efficiency, for a given flow is shown in Figure 3.4. The cost is dependent on the amount of treatment (i.e. primary only, primary and secondary or primary, secondary and tertiary) used and the size of the units used in the treatment process. Percent Biological Oxygen Demand removal was the parameter used to represent treatment efficiency in this diagram.

The objective function is defined as the sum of both the cost functions associated with the flow variables in the links and the cost functions associated with the flow variables and efficiencies in the treatment plants. Feasibility must also be maintained regarding the

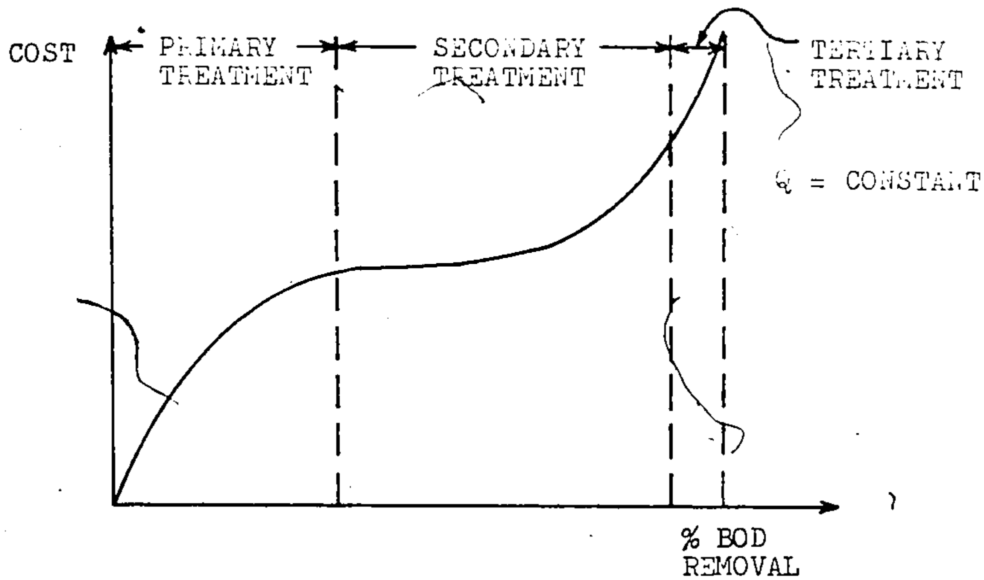


FIGURE 3·4 - TYPICAL PROCESSING COST vs. EFFICIENCY CURVE

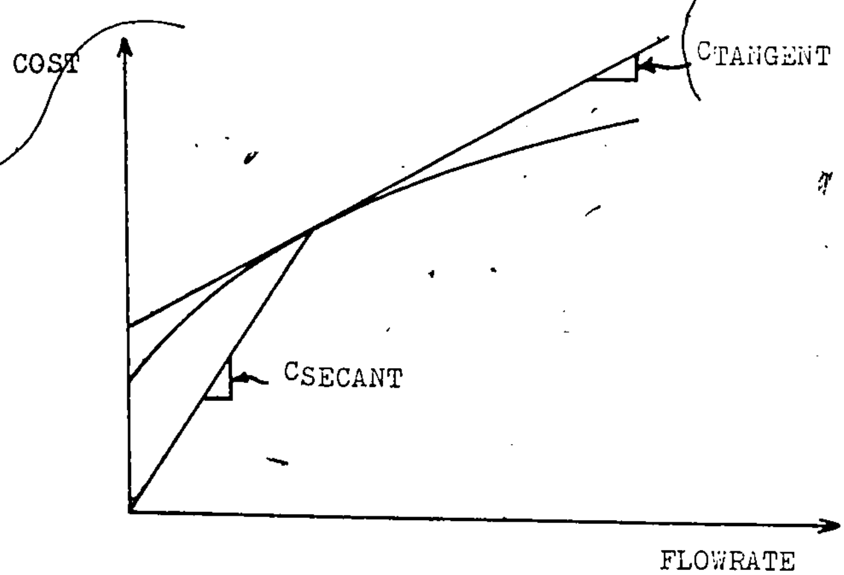


FIGURE 3·6 - CALCULATION OF THE COST COEFFICIENT FOR A TRANSPORTATION LINK USING SECANT AND TANGENT APPROXIMATIONS aformentioned constraints at each of the supply nodes and treatment plants.

The total cost can then be formulated as:

$$\text{TOTAL COST} = z = \sum_{i=1}^{L2} C_i(Q) + \sum_{i=L2+1}^{NQ} C_j(Q,\varepsilon) \qquad (3.6)$$

where,

$C_i(Q)$ = cost function for the i^{th} flow variable,

$C_j(Q,\varepsilon)$ = cost function for the j^{th} flow variable and j^{th} efficiency, and

L2 = total number of flow variables, excluding slack variables.

The objective is to find, from a circuited network, a network which will produce a minimum cost, satisfy the flow constraints at each node as well as satisfying the water quality standards.

This network will be a tree or "branching" network. Figure 3.5 illustrates a branching network which represents a possible solution to the circuited network shown in Figure 1.1. A branching network is characterized by its tree-like structure in which only one link emanates from any supply node or junction. Optimal solutions generally take the form of a branching network due to the economic advantage of transporting a waste from a supply node via one link instead of several. It should be noted that for the solution given

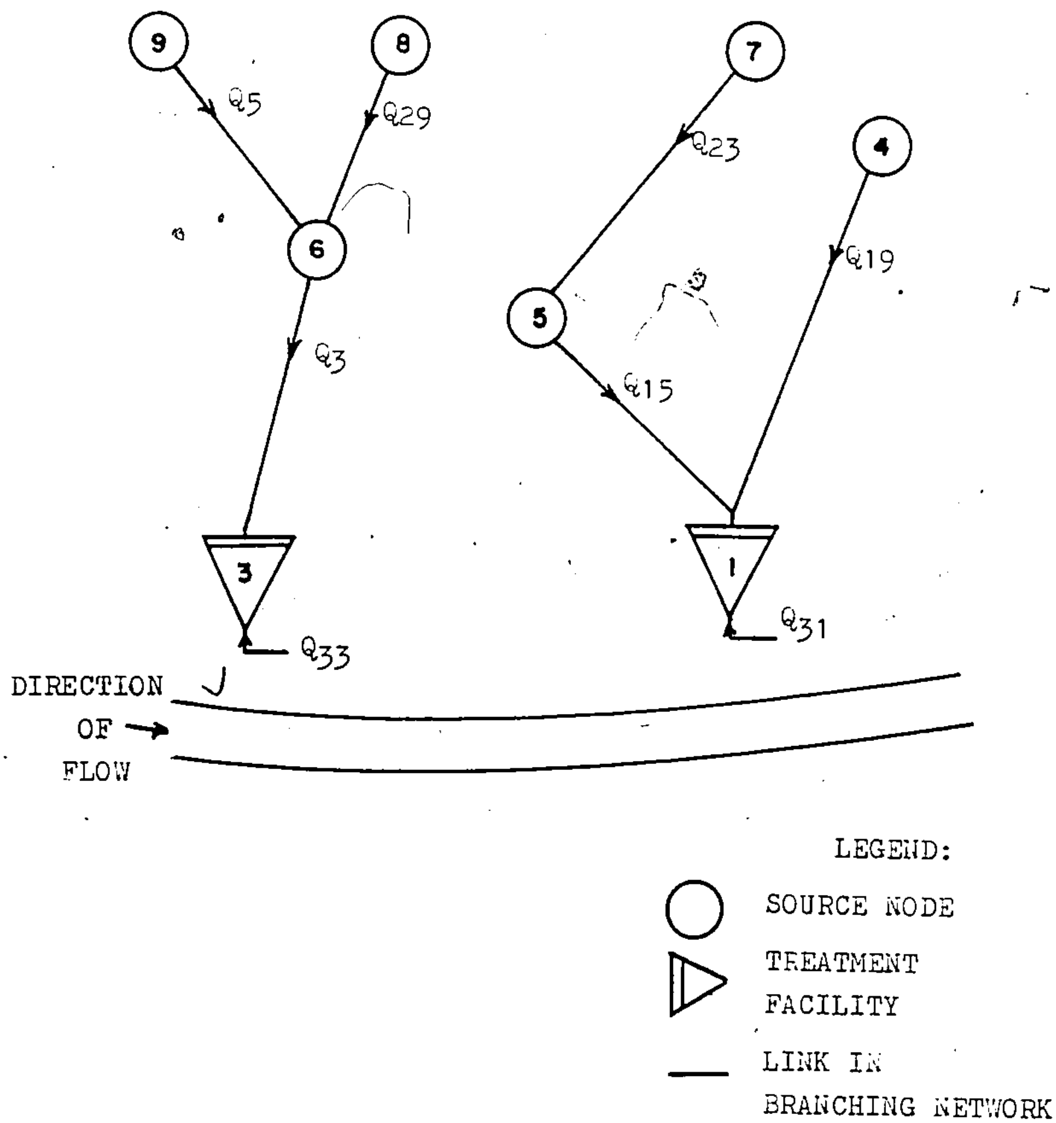


FIGURE 3·5 - BRANCHING NETWORK FOR A REGIONAL SYSTEM in Figure 3.5 that node #2, a treatment facility, has no incoming flow. In this case, the slack variable for node #2 would be set equal to the capacity (stipulation) of node #2. When more than one processing node is possible a network comprising several trees may be formed; one tree is associated with each processing node and they may or may not be connected.

3.3 Cost Coefficients

The cost functions for the transportation links were previously stated to be monotonic, concave and to possess a fixed initial charge. With this kept in mind, a cost coefficient can be found for any given flow. The cost coefficient is equal to the cost for a given link divided by the flow for that link.

There are several methods for calculating the cost coefficient [30]. The use of the secant (see figure 3.6) to approximate the cost curve was used here since the use of the tangent as an approximation necessitates an additional function evaluation and the inclusion of variable correction terms in the objective function.

If the efficiency of each treatment plant is known then a cost coefficient for a treatment facility can be found since the cost vs. flow curve is monotonic and concave.

Recalling that the slack variable is equal to the plant capacity minus the flow through the plant then for the branching network shown in Figure 3.5 the slack variable, Q_{33}, is set equal to $Q_{33} = D_3 - Q_3$. To be consistent the cost coefficient must therefore be calculated as shown in Figure 3.7. Thus $(C_p)_i$ is equal to:

$$(C_p)_i = \left(\frac{COST_{cap} - COST_{plant}}{Q_{cap} - Q_{plant}} \right) \qquad (3.7)$$

However, to ensure that a bounded solution is obtained for finite values of flow the sign of the cost coefficient must be reversed. Therefore $(C_p)_i$ is equal to:

$$(C_p)_i = - \left(\frac{COST_{cap} - COST_{plant}}{Q_{cap} - Q_{plant}} \right) \qquad (3.8)$$

The use of the slack variables allows the processing quantitites and processing costs to be expressed independently of the transporation quantities. This produces a straightforward method for calculating the artificial objective function which may be stated in the following manner.

$$\underset{Q}{\text{Min}}\ Z_{art} = \sum_{i=1}^{NQ-NCEN} (C_t)_i\, Q_i + \sum_{i=L2+1}^{NQ} (-C_p)_i\, Q_i \qquad (3.9)$$

subject to:

$$\bar{A} \times \bar{B} = \bar{Q}$$

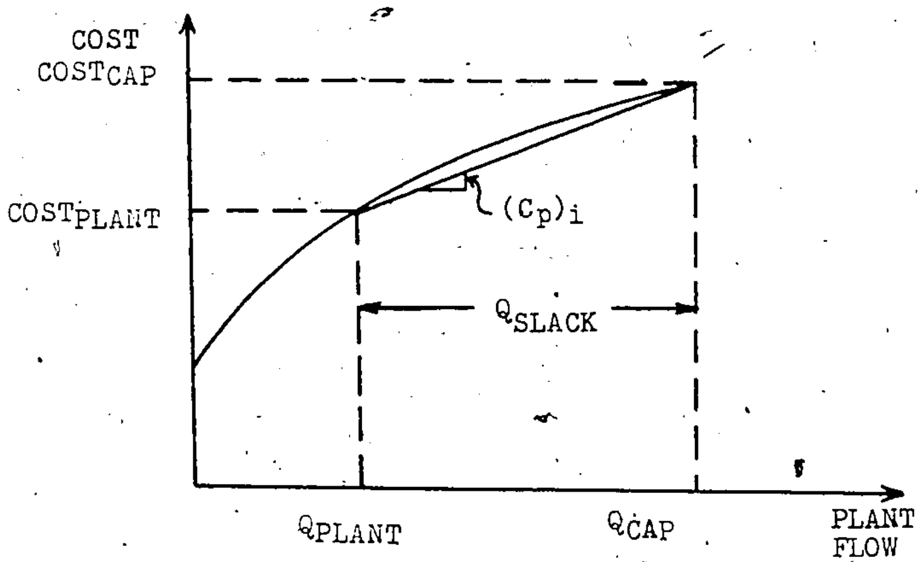


FIGURE 3·7 - CALCULATION OF THE COST COEFFICIENT FOR A TREATMENT FACILITY and

$$Q_i \geq 0 \quad i = 1, NQ$$

where,

$(C_t)_i = f(Q)$

$(C_p)_i = f(Q,\varepsilon)$

$(C_t)_i$ = transportation cost coefficient for the i^{th} flow variable.

$(C_p)_i$ = processing cost coefficient for the i^{th} treatment centre.

NCEN = number of treatment centres.

In order to calculate the true cost for any branching network another set of cost coefficients are stored which are based on the flows in the treatment plants.

The problem of equation 3.9 now becomes one of optimization of a nonlinear objective function subject to a set of constraints which are linear in the design variables.

Other properties of the problem are:

1. The problem is one of minimization,
2. The objective function is nonlinear only insofar as the cost coefficients are functions of the design variables.

3.4 The Mathematical Algorithm

The algorithm given below was used to solve this problem.

1. Assume an initial set of flows Q^o and stipulate the

water quality standard which must be met.

2. Determine compatible slack variables for Q^o.
3. Calculate the cost coefficients, C. The cost coefficients for the links are a function of the current flow values Q^o. From the slack variables determine the flow through each plant. Enter the dynamic programming routine with this set of flow variables and determine the efficiencies for each plant which will satisfy the water quality standard at a minimum cost. Calculate the cost coefficients for the treatment facilities based on the respective slack variables and the treatment efficiency. All cost coefficients are found using secant approximations.
4. Obtain a solution by linear programming using C and yielding an improved estimate of flows, Q'.
5. If the best solution has been found then the vector Q' forms the required solution. Otherwise replace $Q^o = Q'$ and return to step 3.

3.5 Hydraulic and Physical Constraints

So far, the problem has been restricted to a branching network which will minimize the cost and satisfy the flow constraints at each node.

In addition to these flow constraints there are a set of hydraulic and physical constraints which must be satisfied for each link.

The hydraulic and physical constraints are as follows:

1) Minimum pipe cover, necessary to avoid freezing of wastewater in cold climates or excess shock to the pipes from overhead traffic.
2) Minimum allowable diameter; necessary to avoid plugging of pipes (typically 8 in. (200 mm) in North America).
3) Minimum allowable velocity; necessary to achieve a self cleansing action in the sewer.
4) Maximum allowable velocity; necessary to ensure that excessive scouring of the inner walls of the pipe does not occur.
5) Diameter progression constraint; a downstream link must have a diameter greater than or equal to the link directly upstream to it.
6) Invert progression constraint; the invert elevation of a downstream link must be as low as or lower than the invert level of all links directly upstream of the link.

CHAPTER 4

LINEAR AND DYNAMIC PROGRAMMING

4.1 A Linear Programming Algorithm

Problems involving either, maximization of a concave objective function or minimization of a convex objective function, subject to a set of linear constraints are termed convex programming problems. Problems involving either, minimization of a concave objective function or maximization of a convex objective function, subject to a set of linear constraints may be termed concave programming problems.

Linear programming may be used to solve both convex and concave programming problems. In order to solve convex programming problems it is necessary to employ piecewise linearization of the objective function. This, in turn, requires the introduction of extra variables and extra constraints. The distinction between concave and convex programming, and the properties of the final solutions to these two types of problems are dealt with in greater detail in Appendix C.

The type of problem to be dealt with in this study is minimization of a concave objective function subject to a set of linear constraints.

Subroutine SIMPLEX [29] may be used to illustrate the logic within a linear programming algorithm. The four node network in Figure 4.1 will be used as an aid in describing the subroutine. In subroutine SIMPLEX artificial variables are required to generate an initial solution.

The initial values of the cost coefficients for the links and treatment centre are determined as a function of the initial flow values assigned to the problem. The cost coefficients assigned to the artificial variables are set equal to an arbitrarily high value in order that these variables will be driven out of the solution.

The cost equation would then take the following form:

$$COST = C_1Q_1 + C_1Q_2 + C_3Q_3 + C_4Q_4 + C_sQ_s + C_6Q_6 + C_7Q_7 + C_8Q_8 + C_9Q_9 + M_{10}Q_{10} + M_{11}Q_{11} + M_{12}Q_{12} \quad (4.1)$$

where,

C_1 to C_8 = cost coefficients for the transportation links,

C_9 = cost coefficient of the slack variable for node 4, and,

M_{10}, M_{11} and M_{12} = cost coefficients of the artificial variables for nodes 1, 2 and 3, respectively.

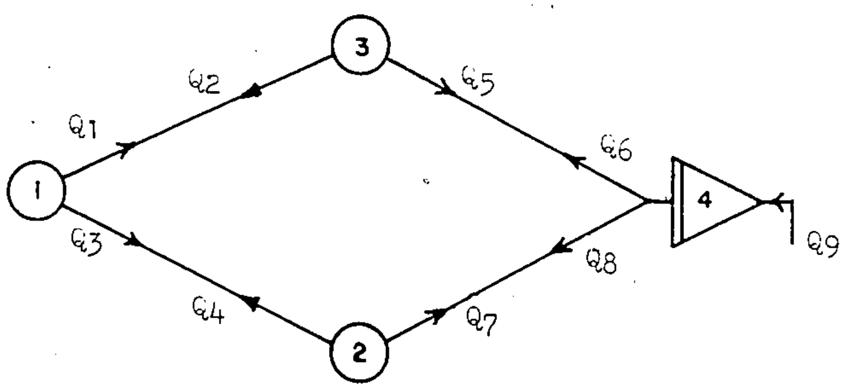


FIGURE 4·1 - FOUR NODE SEWERAGE NETWORK

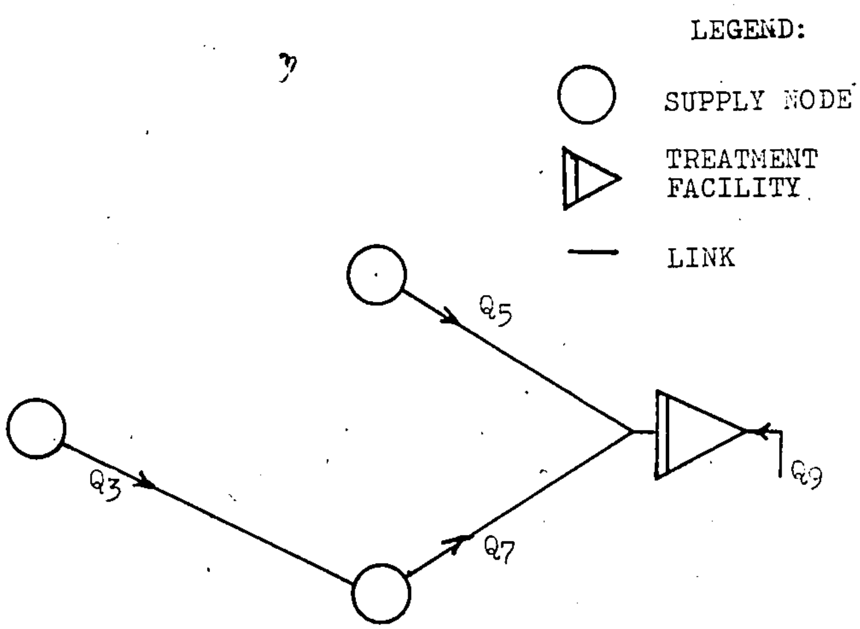


FIGURE 4·2 - INITIAL BRANCHING NETWORK

The set of cost coefficients are used as inputs into subroutine SIMPLEX in order to calculate an updated set of flow variables.

Any flow value with a value greater than zero is termed a basic variable. All other variables are called nonbasic variables. The basic variables form a basic feasible solution. A basic feasible solution is a solution which lies on a vertex of the feasible region (see Appendix C).

The slack variable and three artificial variables form the initial basic feasible solution.

The basic variables will be replaced by nonbasic variables one at a time. If it is assumed that Q_2, Q_4, Q_6 and Q_8 will not replace the artificial variables due to the nature of the problem then Q_{11} will be replaced by Q_7. This variable is the only variable which will satisfy continuity at node 2. Similarly, Q_{12} will be replaced by Q_5. At node 1 both Q_1 or Q_3 could satisfy the continuity constraint and thus either nonbasic variable could replace Q_{10}.

Each of these two nonbasic variables has associated with it a simplex coefficient. The simplex coefficient relates the net change in the objective function per unit of a nonbasic variable hypothetically introduced into the basic feasible solution. The variable causing the greatest improvement is selected to be the incoming variable.

The simplex coefficient for the nonbasic variable Q_j is given by the expression

$$\Delta z_j = C_j - \sum_{i=1}^{NQ} A_{ij} * C_i = C_j - z_j \qquad (4.2)$$

where,

Δz_j = the value of the simplex coefficient for the nonbasic variable Q_j and represents the net change in the objective function by making $Q_j = 1$,

C_j = the cost coefficient of the nonbasic variable Q_j,

C_i = the cost coefficient of the basic variable Q_i, and

A_{ij} = the structural coefficient of the i^{th} row and j^{th} column.

If it is assumed that Q_3 has the largest negative simplex coefficient then the first branching network found by subroutine SIMPLEX would take the form shown in Figure 4.2. Variables Q_3, Q_5, Q_7 and Q_9 now form the new basic feasible solution. The cost coefficients for the flow variables Q_1 through Q_9 will be recalculated based on the new flow variables and subroutine SIMPLEX will be entered a second time.

The only decision to be made is to see if the replacement of Q_3 with Q_1 would result in a lower cost. This will be determined by calculating the simplex coefficient for Q_1, a nonbasic variable. A negative value for the simplex coefficient would represent an improvement in the solution and Q_1 would replace Q_3. A positive value for the simplex coefficient would imply that replacing Q_3 with Q_1 would not result in an improvement and the final solution would remain unchanged.

4.2 Water Quality Parameters

For a given set of treatment plant flows there are several combinations of treatment plant efficiencies that will satisfy a specific water quality standard. The goal is therefore, to select the efficiencies of each of the treatment plants that will satisfy the water quality standard at a least cost. This is the objective of the dynamic programming model. As will be shown later, there is an interdependence between the efficiencies selected at each plant. Dynamic programing is ideal for this type of problem since it can easily account for this interdependence.

Water quality is measured using a number of physical, chemical and biological characteristics. Only the carbonaceous biological oxygen demand (BOD) of wastewater and its effect on the stream dissolved oxygen (DO) will be considered in this study. Dissolved oxygen has long been recognized as a prime indicator of water quality. Adequate levels of DO are necessary to support fish life in a river and to maintain aesthetically pleasing conditions. The model presented herein, could, with modifications, be used with any other quality measure. Whatever parameter is used, it should be ensured that other wastewater constituents be accounted for by fixing their allowable discharge levels at the treatment plants.

There are several mathematical models which predict the effect of BOD discharges on DO. The model used in this study employs the Streeter-Phelps equations [32].

The Streeter-Phelps equations state that the dissolved oxygen concentration in a stream is governed by two reactions: a) the oxygen is depleted by the respiration of bacteria stabilizing organic matter, and b) the oxygen is replentished by rearation from the atmosphere at the water surface. Both of these reactions are assumed to obey first order kinetics. The BOD and DO are decreased at a rate proportional to the remaining BOD. The DO is increased at a rate proportional to the oxygen deficit, D, where the oxygen deficit is defined as the difference between the actual oxygen concentration of the water and the saturation concentration, D_s, at the water temperature.

Other more complicated models, such as Dobbin's [11] model were not considered in this study since the assignment of values to the additional parameters which are required in this model would have been arbitrary.

The BOD decay equation, for any time t is commonly written as follows:

$$L = L_o e^{-kt} \tag{4.3}$$

The Streeter-Phelps equation for the oxygen deficit at any time t is:

$$D = \left(\frac{k}{r-k}\right)(e^{-kt} - e^{-rt})L_o + (e^{-rt})D_o \tag{4.4}$$

where,

L = BOD concentration at time t,

D = oxygen deficit at time t,

k = deoxygenation rate constant,

r = reoxygenation rate constant,

and at time t = 0, $L = L_o$ and $D = D_o$.

This model assumes that the flow is steady and that there is no longitudinal dispersion.

4.3 A Dynamic Programming Model

The river is divided into n reaches with a potential treatment facility located at the upstream end of each reach. Reach i is defined as the stretch of river between the i^{th} and $(i+1)^{th}$ potential discharge.

Several parameters must be known for each reach. These include, the treatment plant flow $(Q_p)_i$, the BOD concentration at the inlet of the plant $(X_{in})_i$, the time of travel for the reach $(T_r)_i$, the dissolved oxygen deficit of the wastewater $(T)_i$, the reoxygenation coefficient $(r)_i$ and the deoxygenation coefficient $(k)_i$. These parameters may vary for each reach. In addition, the river flow $(Q_r)_o$ and the BOD concentration $(F)_o$ immediately upstream of the first potential plant, the stream dissolved oxygen saturation concentration and the minimum allowable dissolved oxygen concentration must be given (see Figure 4.3). By using these parameters and the set of specified treatment efficiencies at each treatment plant it is possible to determine the water quality in any reach.

The dissolved oxygen deficit at the end of the reach is the water quality parameter which is used in this study. The user, by specifying the saturated dissolved concentration and the minimum allowable DO concentration defines the maximum allowable dissolved oxygen deficit, DALLOW. The maximum allowable oxygen deficit can then be used to define a range of possible levels of water quality (oxygen deficit). This range can then be divided into a number of increments which represent discrete levels of water quality.

At the uppermost reach there is only one way of obtaining a specific level of water quality. There are, however, several different combinations of levels of water quality in the first reach and treatment efficiencies in the

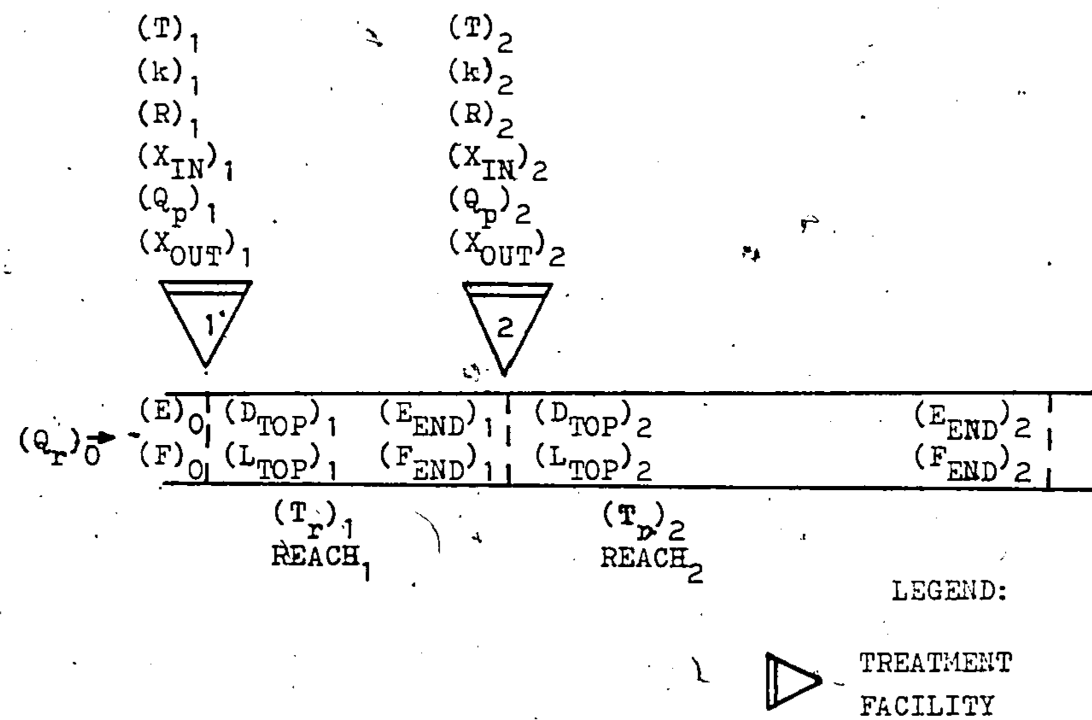


FIGURE 4·3 - REQUIRED PARAMETERS FOR DYNAMIC PROGRAMMING MODEL second reach which will result in the same level of water quality being attained in the second reach. A table can be constructed which is comprised of discrete levels of water quality in the second reach and the total minimum cost of attaining these qualities. A record of the combinations of efficiencies at plants 1 and 2 which led to these minimum costs should be kept.

Similarly, for the third reach there are several ways of attaining a specific level of water quality. Each of these ways involves a particular level of water quality in reach 2 and a specific degree of treatment at plant 3. The various combinations of treatment efficiencies for plants 1 and 2 which led to the level of water quality in reach 3 do not have to be investigated again since the minimum cost for attaining each discrete level of water quality in reach 2 has already been tabulated.

Therefore, for reach i, a new table may be constructed which contains the discrete levels of water quality and the associated minimum costs. A minimum cost for reach i is found by combining the minimum cost for reach (i-1) with the treatment efficiency of plant i which resulted in this minimum cost.

When the levels of water quality have been calculated for all reaches the total minimum cost and treatment efficiencies for all plants can be determined by working backwards through the tables. The total minimum cost and the treatment efficiency for plant N which resulted in this minimum cost can be obtained from the last table. The level of water quality which led to this minimum cost can then be found for reach (N-1). The treatment plant efficiency and minimum cost for reach (N-1) can then be determined. This procedure is repeated until all the treatment efficiencies have been obtained.

This method of solution is referred to as dynamic programming. The total number of combinations which would have to be investigated is significantly less than if complete enumeration was used.

The BOD decay equation and the oxygen deficit equation may be used as a basis to develop the set of equations which are necessary to fully define the levels of water quality for each reach of the river. For example, if the level of water quality at the end of reach 1 is to be found then the following steps are necessary (see Figure 4.3).

1. Select the treatment efficiency

$$e_1 = 1 - \left(\frac{X_{OUT}}{X_{IN}} \right) \tag{4.5}$$

2. Calculate the BOD concentration at the top of reach 1.

$$(L_{TOP})_1 = [(Q_r)_o \times (F)_o + (X_{OUT})_1 \times (Q_p)_1] / [(Q_r)_o + (Q_p)_1] \qquad (4.6)$$

3. Calculate the dissolved oxygen deficit at the top of reach 1.

$$(D_{TOP})_1 = [(Q_r)_o \times (E)_o + (T)_1 \times (Q_p)_1] / [(Q_r)_o + (Q_p)_1] \qquad (4.7)$$

4. Calculate the BOD concentration at the end of reach 1.

$$(F_{END})_1 = h_1 \times (L_{TOP})_1 \qquad (4.8)$$

where,

$$h_1 = e^{-(k)_1 \times (T_R)_1} \qquad (4.9)$$

5. Calculate the dissolved oxygen deficit at the end of reach 1.

$$(E_{END})_1 = f_1 \times (L_{TOP})_1 + g_1 \times (D_{TOP})_1 \qquad (4.10)$$

where,

$$f_1 = \frac{k_1}{R_1 - k_1} \left(e^{-(k)_1 \times (T_R)_1} - e^{-(R)_1 \times (T_R)_1}\right) \qquad (4.11)$$

and,

$$g_1 = e^{-(R_1) \times (T_R)_1} \qquad (4.12)$$

where,

e_1 = the treatment efficiency at treatment plant 1,

$(X_{OUT})_1$ = the BOD concentration of the effluent from plant 1, (mg/l),

$(L_{TOP})_1$ = the BOD concentration at the top of reach 1, mg/l,

$(D_{TOP})_1$ = the oxygen deficit at the top of reach 1, mg/l,

$(F_{END})_1$ = the BOD concentration at the end of reach 1, mg/l,

$(E_{END})_1$ = the oxygen deficit at the end of reach 1, mg/l.

4.4 Assumptions and Limitations of the Dynamic Programming Model

Equations 4.6 and 4.7 assume complete mixing at the point where the treatment plant effluent enters the river. The values obtained for $(E_{END})_1$ and $(F_{END})_1$ can be used in the calculation of $(E_{TOP})_2$ and $(F_{TOP})_2$.

For a particular set of treatment efficiencies it can then be found if the specified maximum allowable deficit has been exceeded in any reach. If the maximum deficit is exceeded then a very large penalty cost is assigned to the cost for the set of treatment efficiencies. The large penalty cost ensures that the set of efficiencies selected does not result in the minimum cost.

In addition, due to the nature of equation (4.10) a check must be made to ensure that the maximum allowable deficit is not exceeded at any intermediate point in any reach.

There are several assumptions which have been made in this model: It has been assumed that the saturation concentration of oxygen is constant throughout the length of the river. This assumption may not be valid in long rivers where a change in temperature and thus a resulting change in the saturation concentration may occur. However, the problem being studied here concerns urbanized areas and therefore the river lengths are relatively short.

The model could be modified to allow for the saturation concentration of oxygen to vary from one reach to another. An additional matrix which would contain the different values of the saturation concentrations would be required. Also, the method used for calculating the maximum oxygen deficit at any intermediate point within the reach would have to be modified to take into account the changing saturation concentrations of oxygen.

In this model the reaeration capacity of a flowing stream is assumed to depend primarily upon the velocity and degree of turbulence of the stream. Other authors [1] have stated that the rearation capacity is a function of the hydraulic properties of the river which are in turn a function of streamflow. This method may lead to a more accurate measure of the rearation capacity, however additional parameters such as the depth of flow in the channel, roughness of the channel and shape of the channel are required. It was felt that the incorporation of these additional parameters was not necessary since a hypothetical river basin was used in this study. These additional parameters could however be incorporated into the model.

The incorporation of tributaries would require several changes be made to the model. It would be assumed that the tributaries would enter at the top of a reach. The flow in the main river would be adjusted to account for the tributary flow. Complete mixing at the point of entry would be assumed.

Additional bookkeeping would be required to ensure that, before the levels of water quality for a reach were calculated that the levels of water quality for all reaches upstream of this reach were known.

A parameter which is commonly used today to measure the demand of oxygen in a river is total BOD, where total BOD is defined as being the sum of the carbonaceous BOD and the nitrogeneous BOD (the oxygen equivalents of total

Kjeldahl Nitrogen)*. Thus it would be useful to incorporate into the model the oxygen demand due to both carbonaceous BOD and nitrogeneous BOD. Although the decay equations for both of these compounds are first order reactions the deoxygenation coefficients for each compound are different. Therefore additional equations would be required if both nitrogeneous and carbonaceous BOD were considered. Also, the plant efficiencies are selected with respect to a specific parameter. If two parameters were selected then two sets of cost vs. efficiency curves would be required. Cost vs. efficiency curves for nitrogeneous BOD were not readily available. In addition, if two parameters were used the treatment efficiencies would have to be the same unless the program was modified. Use of similar treatment efficiencies would lead to erroneous results since the optimum treatment efficiencies may well be different for each compound. For these reasons only the oxygen demand due to carbonaceous BOD was considered.

If the planner feels that it is necessary to account for the additional oxygen demand due to nitrogeneous BOD then the following procedure could be used. The concentration of Kjeldeahl Nitrogen at the inlet of the

---

* Snodgrass, W., Private Communication on Water Quality Parameters, 1980, Department of Chemical Engineering, McMaster University, Hamilton, Ontario.

treatment plant should be stipulated. This concentration should then be multiplied by 4.57. This factor arises from the assumption that all Kjeldahl Nitrogen decays as ammonia and that 1.0 mg of ammonia consumes 4.57 mg of oxygen, whereas carbonaceous BOD consumes 1.0 mg of oxygen. This concentration would then be added to the concentration of the carbonaceous BOD.

This method will, in part account for the oxygen demand due to nitrogeneous BOD in addition to the demand due to carbonaceous BOD. However, it is only a first approximation as the difference in both the deoxygenation rates and the required treatment efficiencies are not taken into account.

4.5 Assimilative Capacity of the Receiving Body of Water

In attempting to find the final layout and the sizes and locations of the treatment facilities the importance of the autopurification or assimilative capacity of the receiving body of water must be taken into account.

In discussing the dynamic programming model it was pointed out that for a given set of treatment plant flows, the plant efficiencies could be found which would lead to a minimum cost and satisfy the water quality goal. However, at each treatment plant there are a large number of possible flows which may occur. Therefore, there will be a very large number of these sets of treatment plant flows.

Certain sets of flows will lead to poor utilization of the assimilative capacity of the receiving body of water. This will be reflected in high efficiencies being required at several plants. This in turn, will lead to high treatment plant costs (see Appendix D).

The problem is then one of minimizing the transportation and processing costs while simultaneously accounting for the assimilative capacity of the receiving body of water. A qualitative example will be used to illustrate the tradeoff in costs which is required if the least cost system is to be found.

The system shown in Figure 4.4 comprises six supply nodes and three potential treatment plants. A number of alternative transportation routes are shown. The total flow in the network is 50 Imgd.

If only the treatment plant costs and the water quality standards are considered then there would be one set of flows for the treatment plants which would result in the full assimilative capacity of the river being utilized and thus a least cost. The resultant cost would be a function of the individual treatment plant flows and of the plant efficiencies which would be required to satisfy the water quality standards.

As has been previously stated there are many possible sets of treatment plant flows which could lead to

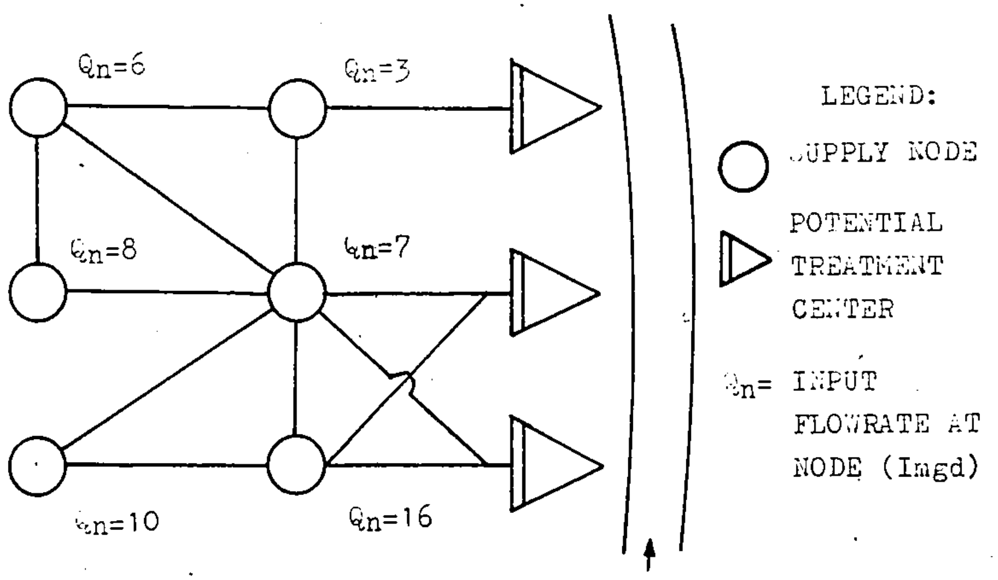


FIGURE 4·4 - SIX NODE SEWERAGE NETWORK

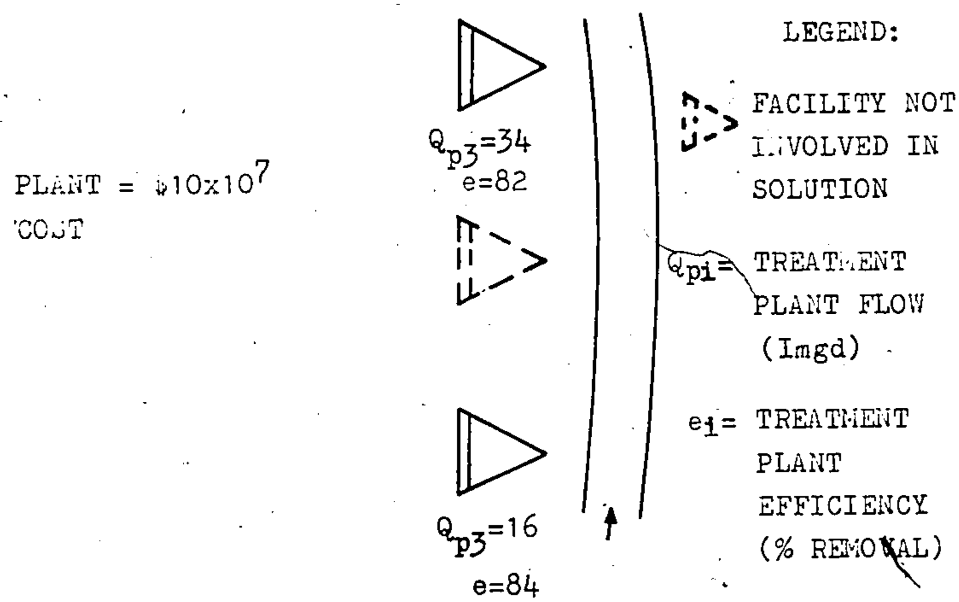


FIGURE 4·5 - A LEAST COST SOLUTION FOR THE TREATMENT FACILITIES this least cost. For this example it will be assumed that the set of treatment plant flows shown in Figure 4.5 results in the least cost system. The hypothetical total cost could be given as \$10 x 10^7.

If the water quality constraints are totally relaxed then the treatment plant efficiencies may be set equal to some arbitrary low value, say 35 percent. This efficiency indicates that only primary treatment has been provided and should ensure the absence of floating solids in the stream. The processing costs would thus be a function of flow only.

The least cost for this type of problem can be found by finding the best combination of the transportation routes and treatment plant sizes and locations. Of the many possible combinations available, assume that the combination shown in Figure 4.6 results in a least cost solution. The link costs were assumed to be \$6. x 10^7, the plant costs \$5. x 10^7.

In combining the two situations, the least cost for the system would be the cost resulting from the best cost of a) the transportation and processing costs and b) the fullest utilization of the receiving body of water. The system shown in Figure 4.7 was assumed to be the final least cost solution. The assumed link costs are \$7. x 10^7, the

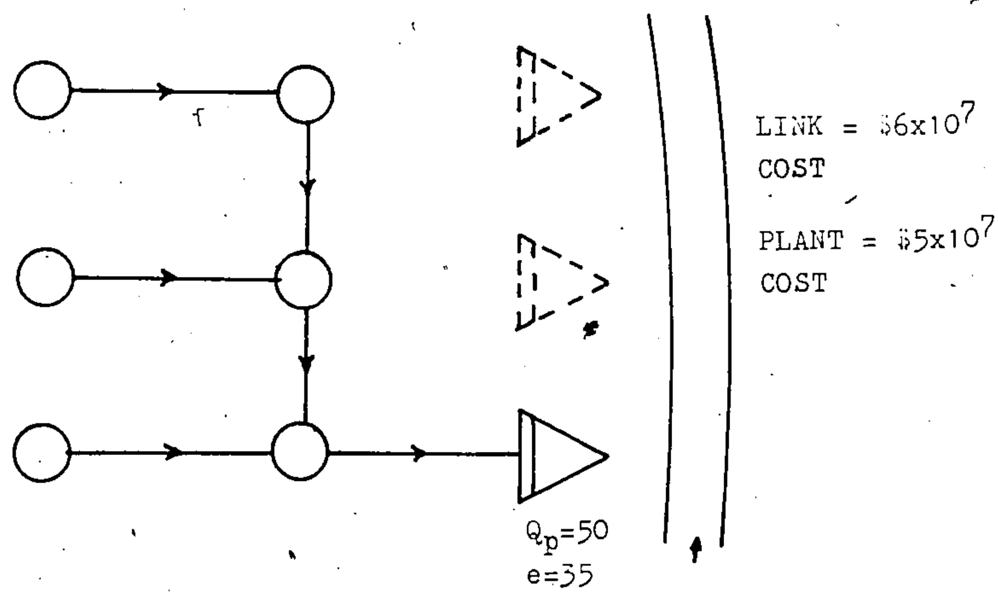


FIGURE 4·6 - A LEAST COST SOLUTION WITH RELAXED WATER QUALITY STANDARDS

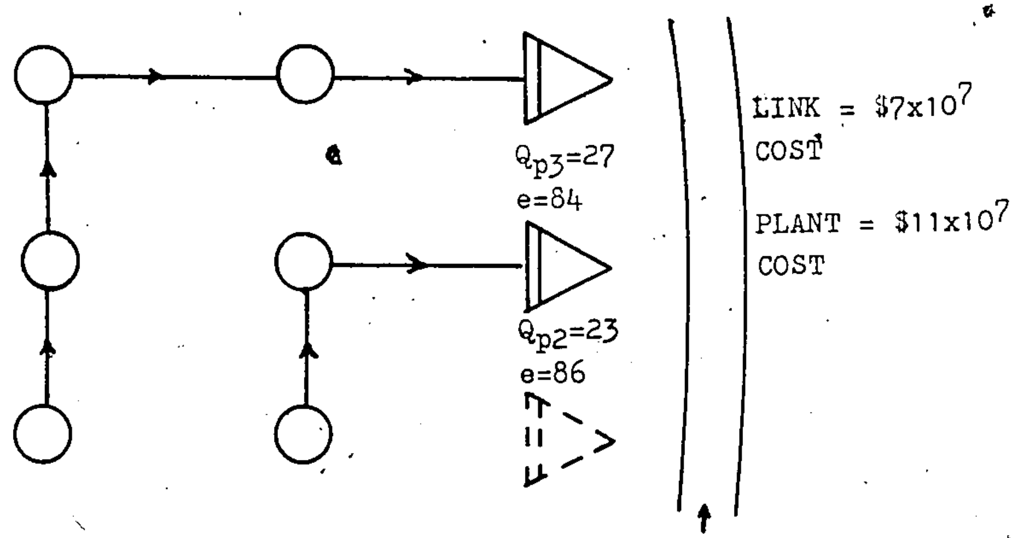


FIGURE 4·7 - A FINAL LEAST COST SOLUTION plant costs \$11. x 10^7.

Comparison of the three final networks shows that additional costs have been incurred. The transportation costs have increased from \$6. x 10^7 (Figure 4.6) to \$7. x 10^7 (Figure 4.7). The processing costs increased from \$10. x 10^7 (Figure 4.5) to \$11. x 10^7 (Figure 4.7). In addition, the full assimilative capacity of the river has not been utlized. This can be seen by comparing the treatment plant costs given in Figure 4.7 and Figure 4.5. The network which results in the least cost (as in Figure 4.7) will be the one which minimizes these additional transportation and processing costs.

CHAPTER 5

THE COMPUTER PACKAGE

5.1 Introduction to the Computer Package

The computer package contains one driver program and 19 associated subroutines. An overall flowchart of the calling sequence for the computer package is illustrated in Figure 5.1.

The computer program has been run primarily in the batch mode. However, it may be advantageous to test smaller networks in the timesharing mode. This user found that smaller problems could be tested in the timesharing mode if a few minor modifications were made in the computer program.

The development of the program has been such that several different variations of the initial circuited network may be easily tested. This may be necessary in order to satisfy practical, social or aesthetic limitations.

The subroutines which calculate the costs for the network were designed to be flexible so that the user may, if he wishes, supply his own cost equations. The restrictions on the shapes of the individual cost curves have been stated in Chapter 3, section 2.

The computer package was designed to require minimal data without reducing the flexibility of the program. The

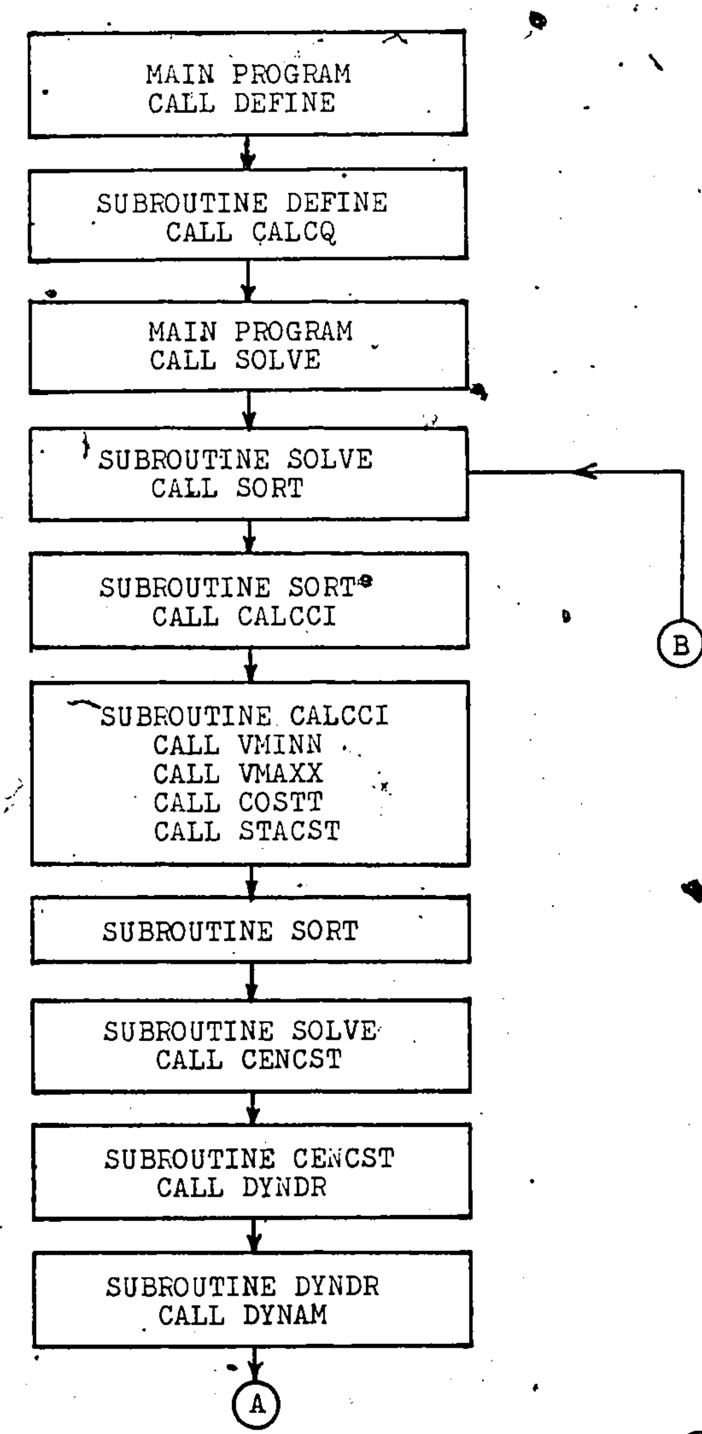
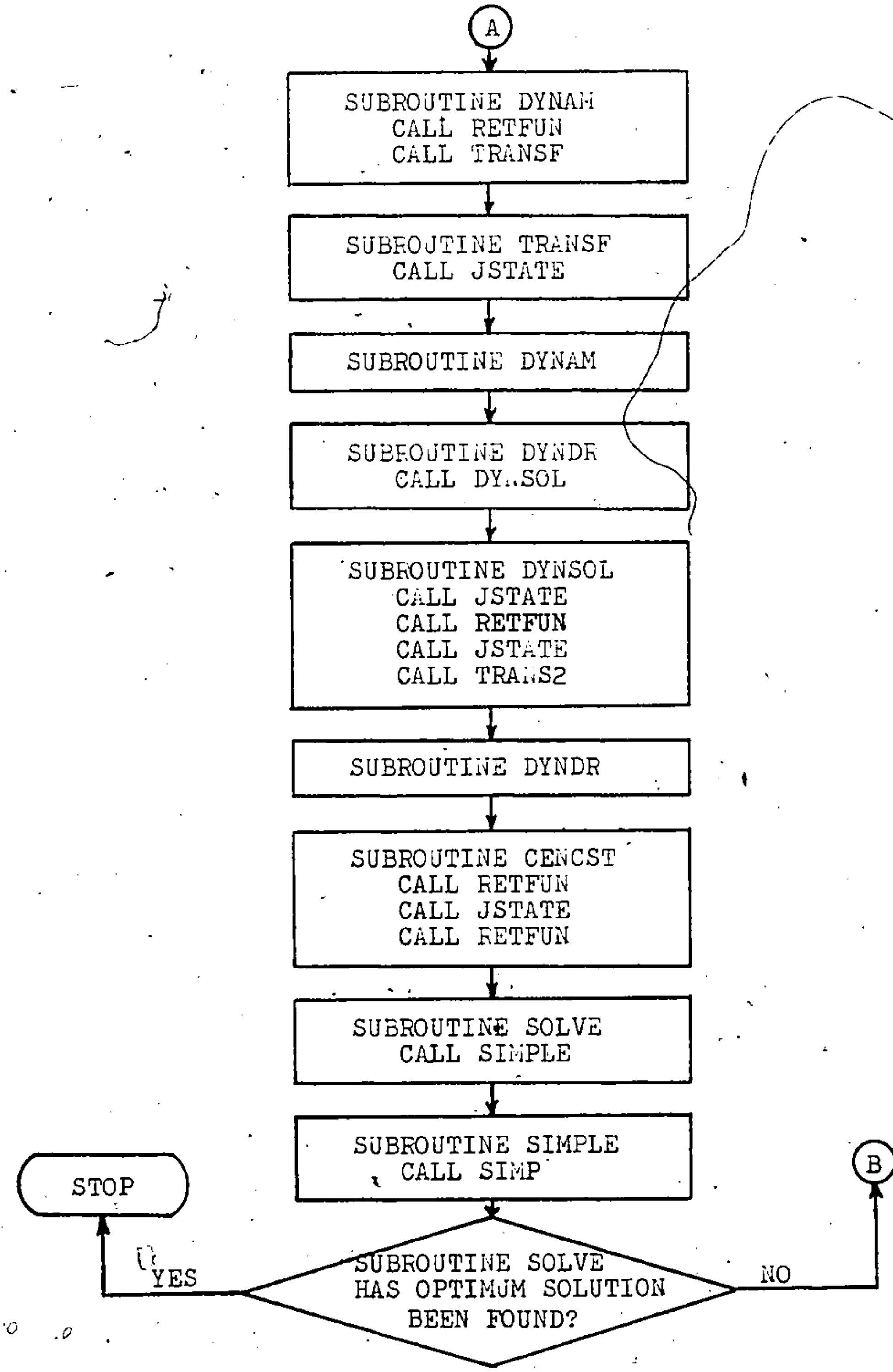


FIGURE 5·1 - CALLING SEQUENCE FOR THE COMPUTER PACKAGE


A
SUBROUTINE DYNAM
CALL RETFUN
CALL TRANSF
SUBROUTINE TRANSF
CALL JSTATE
SUBROUTINE DYNAM
SUBROUTINE DYNDR
CALL DYNSOL
SUBROUTINE DYNSOL
CALL JSTATE
CALL RETFUN
CALL JSTATE
CALL TRANS2
SUBROUTINE DYNDR
SUBROUTINE CENCST
CALL RETFUN
CALL JSTATE
CALL RETFUN
SUBROUTINE SOLVE
CALL SIMPLE
SUBROUTINE SIMPLE
CALL SIMP
SUBROUTINE SOLVE
HAS OPTIMUM SOLUTION
BEEN FOUND?
YES
STOP
NO
B

user must dynamically dimension all arrays in the main program.

5.2.1 The Driver Program

The computer package has been developed so that all arrays are dynamically dimensioned in the driver program. The arrays are then transferred through parameter lists to the relevant subroutines. The allotted array sizes must be sufficient for the specific problem being solved. The necessary sizes of the arrays are given in Appendix E.

A channel number for each of the input, output and data files must be specified within the driver program. Part of the input data is entered directly in the driver program. The remainder is read in through a data file within the driver program. The input data which is necessary to run the program is given in Appendix F. The driver program calls subroutine DEFINE and subroutine SOLVE.

5.2.2 Subroutine DEFINE

Subroutine DEFINE is called by the main program. The purposes of subroutine DEFINE are to read in the node and link data, to determine the minimum and maximum flows at each node and to set up several working arrays. In addition, if the data supplied is incorrect a message will be sent back from subroutine DEFINE.

The arrays containing the commercial and industrial flowrates, the population and the acreage at each node are read in by subroutine DEFINE. Subroutine CALCQ is then called to determine the minimum and maximum design flowrates at each node.

Subroutine DEFINE also calculates the values for all the elements in the structural coefficient matrix, A(N,NQ), and assigns consistent initial values to the slack variables which in turn are based on the initial estimates of the treatment plant flows.

A summary printout of the node and link data may be obtained by setting the logical variable, ECHO equal to TRUE.

Subroutine DEFINE was excerpted from CIVLIB [29]. Substantial modifications were necessary however, in order to apply the subroutine to the specific type of problem being studied here.

5.2.3 Subroutine CALCQ

The purpose of subroutine CALCQ is to calculate the minimum and maximum flowrate at each node. The total flowrate at each node is comprised of any of the following; an industrial flow, a commercial flow, a domestic flow and an infiltration flow.

5.2.4 Subroutine SOLVE

Subroutine SOLVE calls the three subroutines which form the framework for the iterative linear programming algorithm.

For a given set of flowrates subroutine SOLVE calls subroutine SORT and subroutine CENCST. Subroutine SORT returns to suroutine SOLVE a set of cost coefficients for the transportation links. Subroutine CENCST returns to subroutine SOLVE a set of cost coefficients for the treatment plants. These cost coefficients are then used to find an updated set of flows (subroutine SOLVE calls subroutine SIMPLE and subroutine SIMPLE returns to subroutine SOLVE the updated set of flows.)

It is then determined (see Chapter 6, section 7) if this updated set of flows results in a least cost. If the updated set of flows does result in a least cost then the program is terminated. If not, then the updated set of flows are used to determine a new set of cost coefficients and the linear programming iterative procedure is repeated.

The user has the option of obtaining the output for each iteration (set NPRINT=1) or for the final iteration only (set NPRINT=0).

5.2.5 Subroutine SORT

Subroutine SORT returns to subroutine SOLVE a set of cost coefficients for the transportation links, for a given set of flow values.

Within subroutine SORT the allowable upstream invert level and minimum allowable diameter for each link are determined. For the first entrance into subroutine SORT the upstream cover and diameter are set equal to minimum values. This is necessary, since at this point no branching network and thus no accurate upstream elevation or diameter constraints have been calculated.

When a branching network has been determined then the invert elevation for a link will be determined by the minimum cover constraint and the downstream invert elevation of the links directly upstream to it. The minimum cover constraint will set an upper limit on the upstream invert elevation. The incoming links determine the lower limit.

Thus, it becomes imperative to provide a routine which will, for any link i, determine if there are any links upstream of link i, and if so, the routine should denote via a flag, whether or not the downstream invert elevation for the upstream links have been determined.

In order to do this a branching network must be chosen from the many possible layouts. This branching network which constitutes one possible solution is sent back from the linear programming routine. The branching network will contain (N-1) links with finite flows (N = number of nodes). The remaining links will have a flow equal to zero and will be called redundant links.

For any given branching network, subroutine SORT determines whether or not a link is furthermost upstream (called upstream link), a non-upstream link or a redundant link.

If link i is an upstream link then the upstream invert elevation is set equal to the upstream ground elevation minus (minimum cover plus diameter of pipe plus clearance).

If link i is a non-upstream link, then the upstream elevation for link i is set equal to the lowest downstream invert elevation for any incoming link.

The diameter progression constraint is also accounted for in subroutine SORT. For all links the variable USDIA is calculated. USDIA represents the smallest diameter which is allowed for any link, due to the diameter progression constraint.

For an upstream link USDIA is set equal to the smallest diameter available. For a non-upstream link USDIA is set equal to the largest diameter from an incoming link.

The upstream invert elevation and value for USDIA for all redundant links will be calculated after all the branched links have been determined. The selection of the upstream invert elevation and diameter chosen for these links will be explained later.

5.2.6 Subroutine CALCCI

Subroutine CALCCI is an organizational routine which calls subroutines VMINN, VMAXX, COSTT, and if required, STACST. The subroutine also calculates SAVAIL, which is the slope corresponding to the minimum cover at the downstream end of the link. SAVAIL thus acts as an initial lower limit since any slope less than SAVAIL would violate the minimum cover constraint at the downstream end.

The smallest diameter allowable, due to the diameter progression constraint can be calculated for a link with gravity flow conditions. This diameter is then used as the starting diameter. The minimum and maximum velocities as well as the cost are then calculated for the link assuming gravity flow. Each time subroutine CALCCI is reentered, for the link, the diameter is increased by one discrete size. The velocities and cost are then calculated for each diameter. This procedure continues until a least cost for the link is found.

The cover at the downstream end for the diameter leading to the least cost is then calculated. If the cover is less than a specified value, (20 + MINIMUM COVER, (ft) in this case), then the optimum cost for this link is assumed to be via gravitational flow. If the cover at the downstream end is greater than 20.0 + MINIMUM COVER then the force main-pump combination leading to a least cost for non-gravitational flow will be found using subroutine STACST.

The above procedure is designed to save computation time and is based on the fact that pumping stations will not be required until certain excess excavation costs accrue.

The value of 20 feet is only a suggested value and may be altered if the user so desires (the value must be changed directly in the subroutine CALCCI).

The two costs using the different types of flows will be compared and the lowest cost will be selected. Subroutine CALCCI will then return, to subroutine SORT, the minimum cost for each link, the corresponding diameter and total pump horsepower if applicable. The downstream pipe elevation will also be returned.

5.2.7 Subroutine VMINN

Subroutine VMINN is called by subroutine CALCCI.

Subroutine VMINN will find, for a given diameter, the minimum slope, SC, which will provide a carrying capacity QS and will be as self cleansing as a sewer flowing full with velocity VMIN. QS is defined as the minimum design flow at the beginning of the design period. VMIN is defined as the minimum full flow velocity found in the pipe. VMIN is calculated using the critical shear stress approach. The value of VMIN will increase as the diameter increases.

The sewer capacity, QF, is found by multiplying VMIN by the full flow area. The ratio of QS/QF is then used to calculate SC. SC is then compared to SAVAIL. If SC is less than SAVAIL then S = SAVAIL and S will be returned to subroutine CALCCI. This is necessary, so as the minimum cover constraint is not violated.

If SC is greater than SAVAIL then S = SC and S will be returned to subroutine CALCCI. This is necessary, so as to provide the required degree of self cleansing. The value of S will be used in subroutine VMAXX as a starting value.

5.2.8 Subroutine VMAXX

Subroutine VMAXX is called by subroutine CALCCI. Subroutine VMAXX checks to ensure that, for the given values of flow, diameter and slope, the velocity in the pipe is less than a specified value, VMAX. VMAX is chosen by the user.

The starting value for the slope, S, is sent down from subroutine VMINN. QFULL, the full flow, is calculated using the initial slope and the Mannings equation. QFULL divided by the full flow area gives VFULL.

VMAXP is then calculated. VMAXP is dependent on the ratio of QS/QFULL, where QS is defined as the maximum flow at the end of the design period.

If the value of QS/QFULL is less than 1.0 then the pipe will be flowing partially full. VMAXP will be calculated under these conditions.

If VMAXP is less than or equal to VMAX then the maximum velocity constraint has been satisfied and the required slope will be S. If VMAXP is greater than VMAX then S will be set equal to S2. S2 will be reduced by ten percent subject to S2 remaining greater than SMNLMT. SMNLMT is the minimum slope required for sufficient self cleansing at the beginning of the design period.

VMAXP will be recalculated using S2. A check will be made to ensure that QS/QFULL is still less than or equal to 1.0. If QS/QFULL is not less than 1.0 then this diameter cannot satisfy both the capacity requirement and maximum velocity constraint. A default value will be used to ensure that this diameter cannot enter the solution.

If QS/QFULL is less than or equal to 1.0 then VMAXP will again be compared to VMAX. If VMAXP is less than or equal to VMAX then the maximum velocity constraint has been satisfied and S2 will be the required slope. If the link is not an upstream link then a drop manhole would be required at the upstream end to account for the reduction in slope.

If VMAXP is still greater than VMAX then the slope will again be reduced by ten percent and the above procedure will be repeated.

Figure 5.2(a) illustrates the above procedure.

If the value of QS/QFULL is greater than or equal to 1.0 using the initial slope, S, found in subroutine VMINN then the pipe will be flowing full. The slope must be recalculated based on QS as opposed to QFULL.

If VMAXP (VFULL) is found to be less than or equal to VMAX then the maximum velocity constraint has been satisfied and the slope, S, is the required slope.

If VMAXP is greater than VMAX then this pipe diameter will not satisfy both the carrying capacity requirement and the maximum velocity constraint. A default value will be used to ensure that this diameter cannot enter the solution.

Figure 5.2(b) illustrates the above condition.

5.2.8 Subroutine COSTT

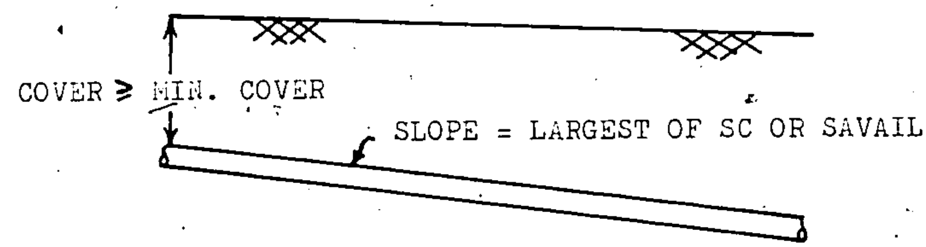
Subroutine COSTT is called from subroutine CALCCI. The subroutine returns the total cost for a link with gravity flow. If the cost calculated is found to be the minimum cost for the link then subroutine COSTT also sends back the downstream invert elevation and the optimum diameter resulting in the minimum cost.

The cost is made up of the cost for the pipe, excavation costs and the cost for the manhole. The manhole is assumed to be at the upstream end of the link.

There are several variables which must be known in order to calculate the cost for a link. The upstream and downstream ground elevations, the upstream and downstream invert elevations are all required to calculate the excavation costs. The diameter must be known to calculate the pipe costs. The invert elevation of the lowest pipe entering the link must be known in order to calculate the manhole costs.

For link i, the smallest diameter, d_1, which

IN SUBROUTINE VMINN



IN SUBROUTINE VMAXX

ASSUME: QS/QFULL<1.0 AND VMAXP≥ VMAX
S2 WILL BE REDUCED SUBJECT TO
S2 ≥ SMNLMT

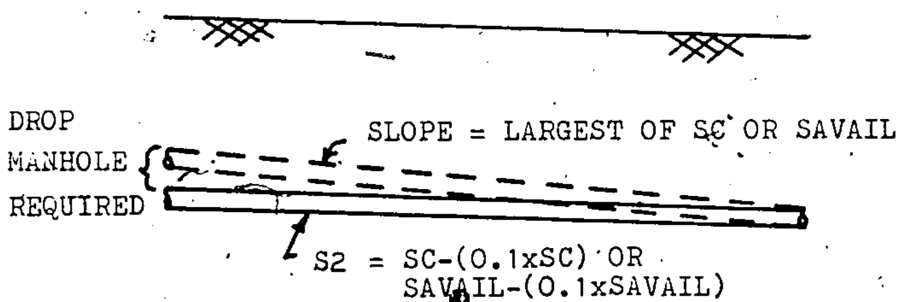


FIGURE 5·2(a) - DETERMINATION OF REQUIRED SLOPE WHICH SATISFIES MINIMUM AND MAXIMUM VELOCITY CONSTRAINTS

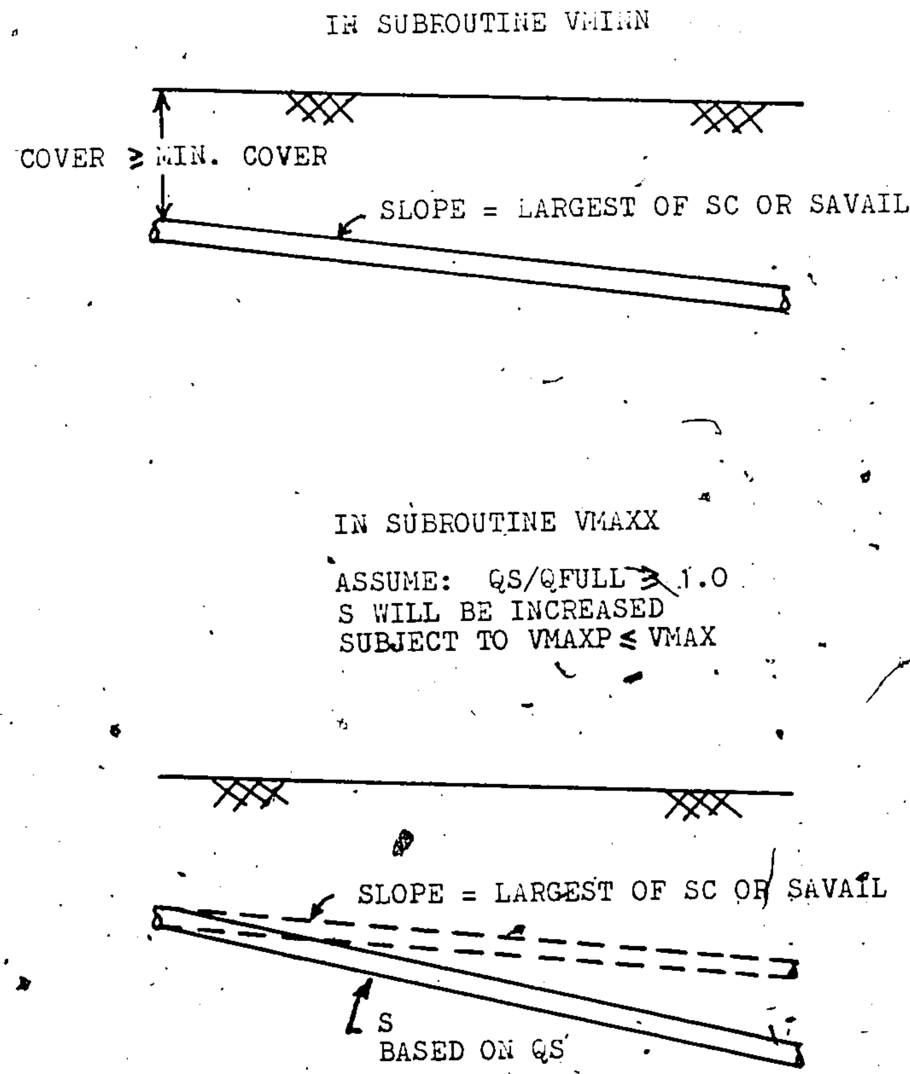


FIGURE 5·2(b) - DETERMINATION OF REQUIRED SLOPE WHICH SATISFIES MINIMUM AND MAXIMUM VELOCITY CONSTRAINTS

satisfies the hydraulic and physical conditions will enter subroutine COSTT first. The cost for link i will be calculated based on d_1. The cost, downstream invert elevation and diameter are then stored.

The second diameter, d_2, which enters subroutine COSTT will be one discrete size larger than d_1. The cost based on d_2 is calculated and compared to the cost based on d_1.

If the cost based on d_2 is greater than the cost based on d_1 then d_1 is assumed to be the diameter which results in the minimum cost for link i. This is based on the assumption that the curve of cost vs. diameter is unimodal (see Appendix A). The values of the downstream invert elevation, EDSOLD, diameter, DOPTOL, and cost, XMNCST, corresponding to d_1 are returned to subroutine CALCCI. DOPTOL and EDSOLD are required for the calculation of the diameters and invert elevations downstream of link i.

If the cost based on diameter d_2 is less than or equal to the cost based on d_1 then the cost, downstream invert elevation and diameter for d_2 are stored.

A third diameter, d_3, would then be used to calculate the cost for link i. The diameter d_3 is one discrete size larger than d_2. The cost based on d_3 is calculated and compared to the cost based on d_2.

The same procedure as above is then repeated. The procedure is terminated when the cost based on d_{i+1} is greater than the cost based on d_i, or when costs based on all available diameters have been found. For the latter case, if the cost based on d_{i+1} is less than the cost based on d_i, then the largest available diameter would be assumed to be the diameter resulting in the minimum cost for the link.

5.2.10 Subroutine STACST

The routine optimizes the distribution of costs between the pipe, pump, electricity and operation and maintenance costs for a known flow rate and Delivery Head (see Appendix G). The cost of the pumping station was then calculated to obtain the total cost for the pumping station. Subroutine STACST returns to subroutine CALCCI, the diameter, PMPDIA, maximum horsepower, HPI, downstream elevation, EDSPMP, which result in the minimum cost, XCOST.

The minimum flow rate, QMIN, and the maximum flow rate, QMAX, are known. Thus the velocities in the pipe for the maximum and minimum flow rates can be calculated using full flow conditions. If the minimum and maximum velocities are both found to be within the velocity limits, which are set by the user, then the total cost for a given diameter will be calculated.

If the maximum velocity constraint is violated for a given diameter, then the diameter is increased to the next largest discrete size. No cost calculations are made. A typical value for the maximum allowable velocity in a steel force main is 15.0 feet per second (fps).

If the minimum velocity constraint is violated for a given diameter, then the discrete diameter just smaller than this diameter is assumed to be the optimal diameter. The reasons ,for doing this will be explained below. The critical shear stress theory was used to calculate the minimum allowable velocity.

The variation of cost as a function of diameter is shown for a typical installation (Figure 5.3). The cost includes the cost of the force main, pump, electricity and operation and maintenance costs.

Deb [10] suggests that all such curves are unimodal. Thus, any comparison of the cost for two discrete diameters will determine whether the diameter leading to the optimal cost has been found.

By starting at the smallest allowable diameter and calculating the costs for diameters which are continuously increasing then, if d_{i+1} is the discrete diameter just larger than d_i and if COST (d_{i+1}) is greater than COST(d_i) then d_i is the diameter leading to the minimum cost.

This procedure is analogous to the procedure used to find the diameter leading to the minimum cost with gravity flow in subroutine COSTT.

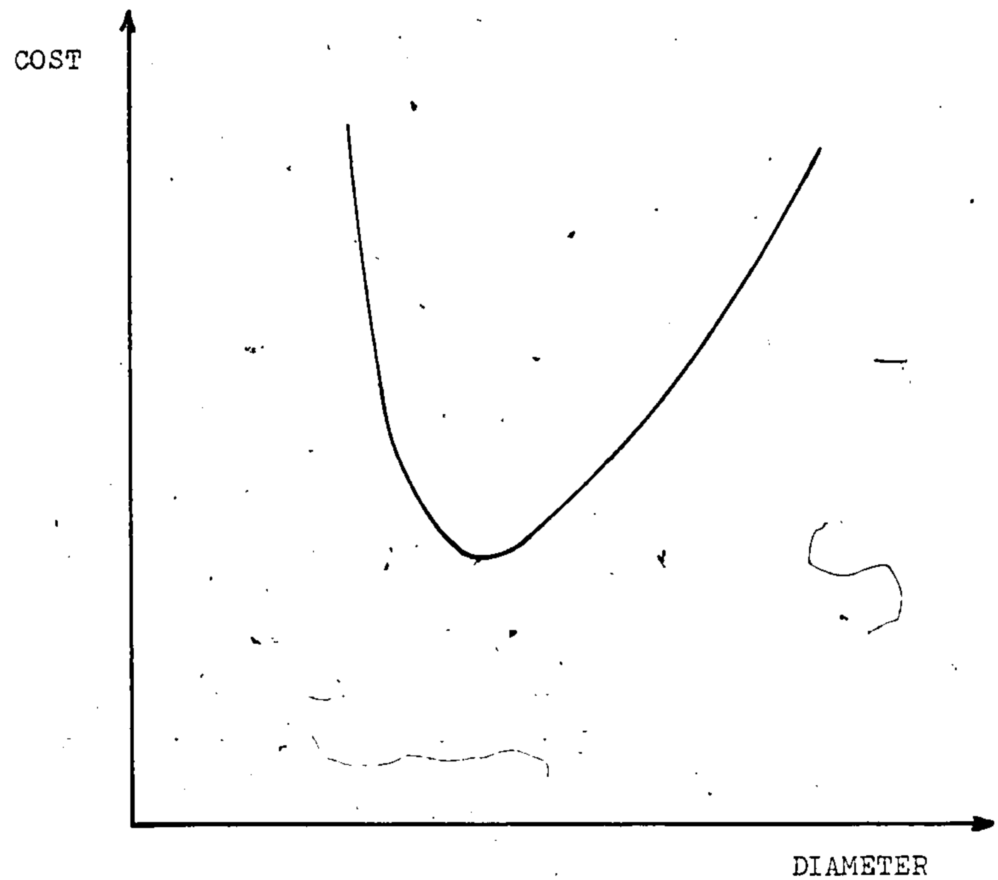


FIGURE 5·3 - COST vs. DIAMETER
(FORCE MAIN- PUMP SYSTEM)

5.2.11 Subroutine CENCST

Subroutine CENCST is called by subroutine SOLVE. This subroutine calculates, for known plant flows and treatment efficiencies the cost coefficient for each treatment plant.

Conversion factors are used to ensure that the flow units are consistent. The procedure used to determine the cost coefficients for the treatment facilities is outlined in Chapter 3, section 3.

5.2.12 Subroutine DYNDR

Subroutine DYNDR is an organizational routine which calls subroutines DYNAM and DYNSOL.

Subroutine DYNDR transfers among many parameters, the individual treatment plant costs, (array FSOL), and the treatment efficiencies, (array DSOL), which correspond to the minimum cost. These two arrays are transferred from subroutine DYNSOL to subroutine CENCST.

5.2.13 Subroutines DYNAM and TRANSF

An overall description of the dynamic programming model used in this study is given in Chapter 4, section 3.

The dynamic programming model starts at the uppermost reach and proceeds in the downstream direction. For each reach a table is produced. This table contains the discrete levels of water quality for the reach and the total minimum cost to obtain these levels. A record is kept of the treatment plant efficiencies and respective costs which resulted in these minimum costs. The water quality variable used is the oxygen deficit at the end of each reach. When the levels of water quality have been calculated for all reaches the total minimum cost can then be found. The treatment efficiency and respective cost for the furthermost downstream plant is then determined. By working upstream the remaining treatment efficiencies and plant costs may be calculated.

In order to obtain the minimum cost for each level of water quality at the end of a reach, i, it is necessary to combine every feasible level of water quality in reach (i-1) with each possible treatment efficiency in reach i. Subroutine TRANSF is called by subroutine DYNAM for each possible combination. Subroutine TRANSF, using the equations outlined in Chapter 4, section 3, then determines the resulting level of water quality at the end of reach i. This value is returned to subroutine DYNAM.

The total cost for this combination, FD(JDS), of water quality in reach (i-1) and treatment efficiency in reach i is then calculated in subroutine DYNAM. This cost is compared to the current lowest cost, FMAX(JDS), which resulted in the same level of water quality being attained at the end of reach i. If FD(JDS) is less than FMAX(JDS) then the lowest cost will be updated and set equal to

FD(JDS). The treatment plant efficiency which resulted in the lowest cost for reach i, DOPT(I,JDS), is also stored. Levels of water quality which were not attainable by any combination will be assigned a high cost. This will ensure that these levels of water quality are not part of the final solution.

It is also necessary to store the BOD concentration at the end of reach i for each level of water quality at the end of reach i. The final values stored are the BOD concentrations which correspond to the optimum plant efficiencies for reach i, (i.e. DOPT(I,JDS)). These values, $(OPTF)_i$ are then used to calculate $(BODTOP)_{i+1}$.

It should also be noted that within subroutine TRANSF a check is made to ensure that the maximum allowable oxygen deficit is not exceeded anywhere within a reach. Any water quality and treatment efficiency which results in the maximum deficit being exceeded at any point along the reach will not be part of the final solution.

5.2.14 Subroutine RETFUN

Subroutine RETFUN is called by subroutine DYNAM. Subroutine RETFUN returns to subroutine DYNAM the treatment plant cost, for a given flow and treatment efficiency.

For a given treatment efficiency the cost vs. flow curve is generally concave. Logarithmic equations were fitted to two concave cost vs. flow curves. One logarithmic equation was required to account for the capital cost of building the treatment plant. The other equation accounted for the annual costs which are required to maintain the facility.

The cost vs. efficiency curve, for a given flow, was divided into linear segments. The dependence of cost on efficiency is therefore easily accounted for.

5.2.15 Subroutines DYNSOL and TRANS2

Subroutines DYNSOL and TRANS2 are used in conjunction to find the total minimum cost and the individual treatment efficiencies and plant costs which resulted in this minimum cost.

The lowest costs for each discrete oxygen deficit at the end of the furthermost downstream reach are compared. From this comparison the minimum total cost is obtained. The discrete oxygen deficit and treatment efficiency of the furthermost downstream plant, (reach (NSTAGE)), which correspond to this minimum cost are stored. The plant cost for this facility is then found by calling subroutine RETFUN.

The next step entails finding the discrete oxygen deficit at the top of reach (NSTAGE). Subroutine TRANS2 is used to find this value. The procedure used to find the oxygen deficit at the top of the reach, DEFICT(J), is an iterative one. A discrete value of the oxygen deficit at the top of reach (NSTAGE)is selected. This deficit is combined with the known treatment efficiency of reach (NSTAGE). The oxygen deficit at the end of this reach, DECOMP, for this combination, is then calculated. If DECOMP equals the known oxygen deficit at the end of the reach, DISCE, then the required oxygen deficit, DEFICT(J), is found.

This value is identical to the oxygen deficit at the end of the adjacent upstream reach, reach (NSTAGE-1). The optimum treatment efficiency of plant (NSTAGE-1) which resulted in this oxygen deficit was previously stored and can thus be determined. The cost of plant (NSTAGE-1) is calculated next. The procedure used to find the oxygen deficit at the upstream end of reach (NSTAGE) is then repeated to find the oxygen deficit at the upstream end of reach (NSTAGE-1).

This method is repeated until all the treatment efficiencies and individual plant costs which lead to the minimum cost have been determined.

5.2.16 Subroutine JSTATE

This brief subroutine determines which element of an array is represented by a specific value. The specific value which is to be matched is transferred through the parameter list to subroutine JSTATE. An array is then scanned to determine which element within this array matches the specified value.

This subroutine was excerpted from CIVLIB [29].

5.2.17 Subroutine SIMPLE

Subroutine SIMPLE is used as a means to calculate the objective function at the optimum conditions. In addition, the dimensions for the work arrays are input into this subroutine. Both subroutine SIMPLE and subroutine SIMP (see below) were taken from the OPTISEP [28] library.

5.2.18 Subroutine SIMP

The primary purpose of subroutine SIMP is to perform the iterative method of linear programming (known as the SIMPLEX method).

5.3 Verification

In order to ensure that the computer package is satisfactory in terms of providing correct results several tests were carried out.

The solution found using the computer package for the branching network of figure 7.3 was verified by hand calculations. The answers found by using the computer package and the answers found using hand calculations were identical for all links.

Comparison of results found by hand calculations to results found by the computer package verified that the dynamic programming part of the program provided correct results.

CHAPTER 6

DEVELOPMENT OF MODEL

6.1 Self Cleansing Velocity for Pipes Flowing Full

Currently, there are two approaches used in the designing of self cleansing sewers.

The minimum velocity approach assumes that the self cleansing action will result from a minimum velocity (typically 2.0 fps) when the sewer, regardless of size is flowing full [36].

The critical shear stress approach suggests that the required self cleansing can be assured if a minimum boundary shear is maintained which is sufficiently strong to prevent suspended particles from settling out on the invert. The minimum boundary shear may be defined as the critical shear stress [20].

Both of these approaches assume that the dominant mechanism in achieving the self cleansing action is the boundary shear.

The difference in the approaches is that the minimum velocity approach assumes that since the velocity is only slightly dependent on the hydraulic radius that the effect of the hydraulic radius may be ignored.

Shields [26] found the equation for the minimum velocity to be:

$$V = \frac{C}{N} R^{1/6} \sqrt{B(S-1)\, D_g} \qquad (6.1)$$

where,

V = required minimum velocity

C = constant (in English system, C = 1.486)

N = Mannings roughness coefficient

R = hydraulic radius

B = dimensionless constant with its magnitude depending on the characteristics of the particle to be prevented from settling out

S = specific gravity of the particle

D_g = diameter of the particle.

Yao [39] showed that the relationship between V and R may be significant. It was found that, under full flow conditions and Mannings N = 0.013 that if a minimum velocity of 2 fps was adequate for a 6 in. sewer, the same self cleansing effectiveness can only be maintained by a minimum velocity of 2.5 fps for a 20 in. sewer.

Yao reasoned that since the minimum velocities are based on the consideration of boundary shears that the boundary shear should be used as the basis for designing self cleansing sewers.

The design equation proposed by Yao is:

$$V = \frac{C}{N} R^{1/6} \left(\frac{T_o}{w}\right)^{1/2} \tag{6.2}$$

where,

T_o = minimum or required boundary shear (critical shear stress), and

w = specific weight of wastewater.

Figure 6.1 indicates that to maintain a given boundary shear, the minimum flow velocity must increase with sewer size.

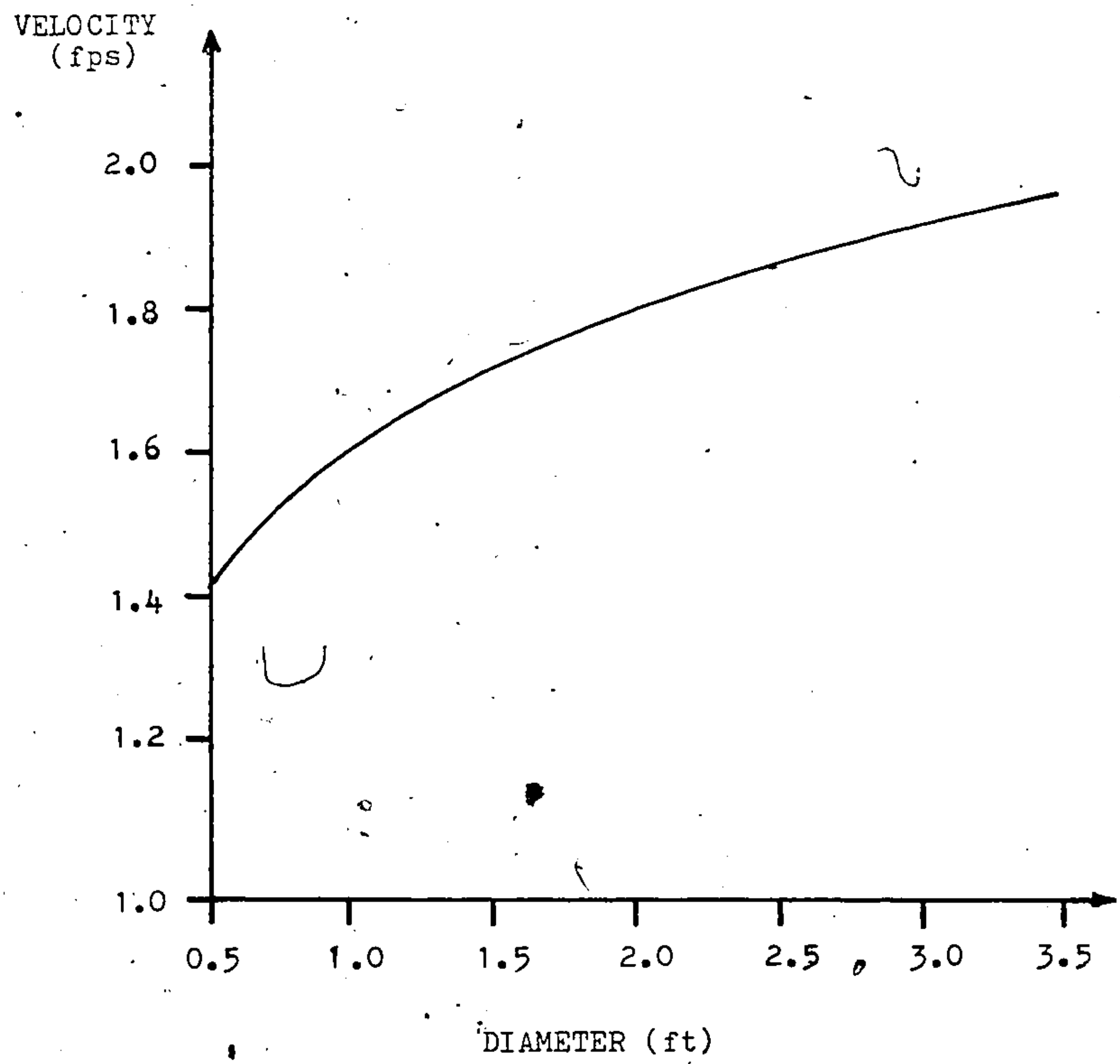
Conversely, if a fixed minimum velocity is used for all sewer sizes the boundary shear will decrease as the sewer size increases. It seems inconsistent to use different boundary shears for one sewer system conveying the same wastewater. Thus the critical shear stress approach is used here.

6.2 Extension to Partial Flow Analysis

Sewer lines are seldom designed to be operated under full flow conditions. Thus, it becomes imperative to provide adequate self cleansing when a sewer is only partially full. The only additional relationship between required full flow and partial flow conditions is that of the average boundary shear stress, T.

The equation:

$$T = \gamma RS \tag{6.3}$$



PARAMETER
VALUES

N = 0.013
To = 0.02 (lb/ft²)
W = 64.8 (lb/ft³)

FIGURE 6·1- MINIMUM FLOW VELOCITY vs. DIAMETER

where,

γ = specific weight of water, and

S = slope of the invert

is applicable to both full and partial full analysis. Thus;

$$\tau = T = \gamma r s_s = \gamma RS \text{ and}$$

$$s_s = (R/r)S \qquad (6.4)$$

where the capital letters denote full flow and small letters indicate partial full flow. The subscript s denotes cleansing equalling that obtained in the full section.

From equation 6.4, the following equation may be derived to compute a slope which ensures equal self cleansing for partially filled sewers.

$$\frac{QS}{QF} = \left(\frac{N}{n}\right)\left(\frac{a}{A}\right)\left(\frac{S}{s_s}\right)^{1/6} \qquad (6.5)$$

in which,

QS = flow rate in partial full sewer

QF = sewer capacity based on V (VMIN)

a = flow area for QS

A = flow area for QF

n = partial flow Manning's n.

The following procedure, from Main [21] was used to find S/s_s. Using Figures 6.2 and 6.3 [35] data points, at intervals of d/D from 0.125 to 0.5, were interpolated from the curves:

d/D vs. QS/QF and

d/D vs. s_s/S

The points were used in a least squares polynomial curve fitting computer program [15] to obtain the following relationships:

QS/QF < 0.19

$$s_s/S = 9.97 - 269.29\,(QS/QF) + 3540.06\,(QS/QF)^2 - 20348.2\,(QS/QF)^3 + 41872.5\,(QS/QF)^4 \quad (6.6)$$

QS/QF $\geq$ 0.19

$$s_s/S = 2.23 - 6.26\,(QS/QF) + 12.46\,(QS/QF)^2 - 13.19\,(QS/QF)^3 + 5.74\,(QS/QF)^4 \quad (6.7)$$

Equation (6.5) shows an influence of a variable ratio of (N/n) on QS/QF.

Yarnell and Woodward [40] and Wilcox [38] in a separate study attributed this variation of n and its effect on the carrying capacity to the variation of roughness in the pipe with depth. Rouse [25] states that, for a given geometry, the secondary currents vary in magnitude with changing depth. This in turn affects the carrying capacity of a sewer for a given depth.

This author was unable to find any confirmatory evidence as to whether the varying roughness or the varying

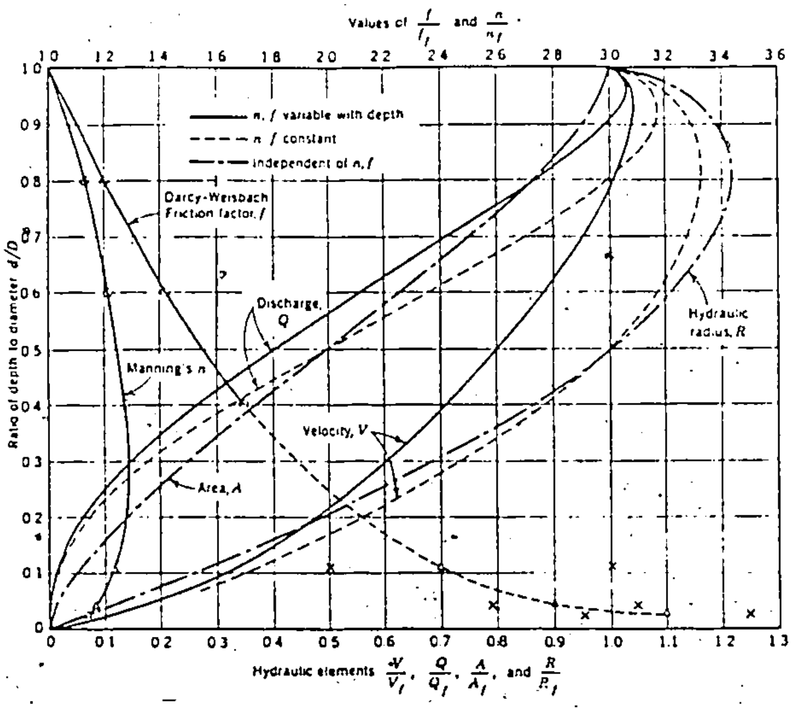


FIGURE 6·2 – HYDRAULICS ELEMENTS GRAPH FOR CIRCULAR SEWERS

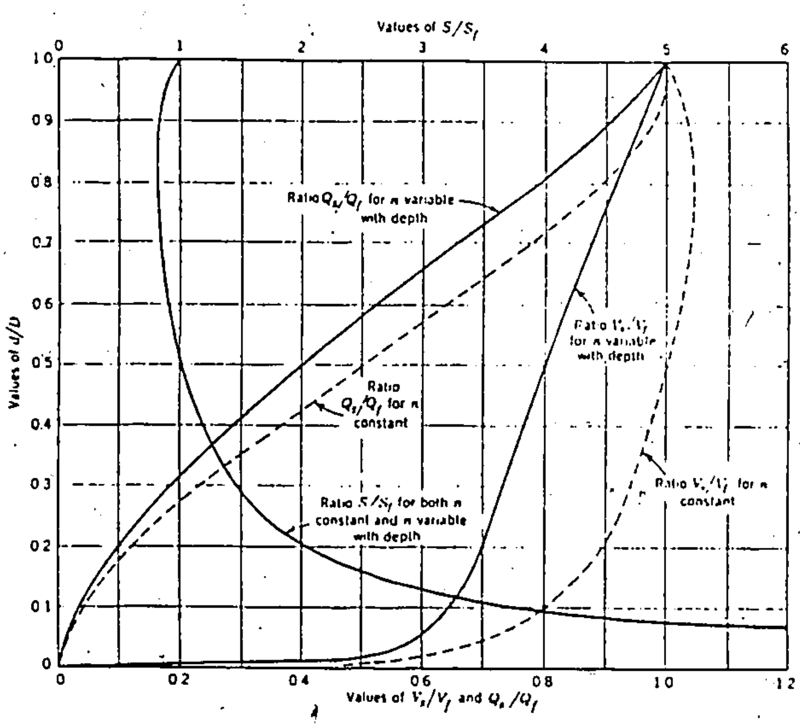


FIGURE 6·3 – HYDRAULIC ELEMENTS OF CIRCULAR SEWERS THAT POSSESS EQUAL SELF CLEANSING PROPERTIES AT ALL DEPTHS secondary currents are responsible for the variation of n with depth.

6.3 Maximum Velocity Constraint

At the end of the design period it should be ensured that excess scouring of the sewers does not occur. To ensure this, the pipe is laid at a slope such that the velocity in the pipe at the maximum flow, VMAXP, is less than a specified value, VMAX. VMAX represents an upper limit to the velocity in the pipe which will not cause scouring. The value for VMAX is determined by the user. A value commonly used is 10 fps.

The procedure used for finding VMAXP was taken from MAIN [21] and is similar to the method used in subroutine VMINN.

The d/D ratio can be obtained from the (QS/QFULL) ratio. The (VMAXP/VFULL) ratio can then be obtained from the d/D ratio. Thus, the velocity VMAXP can be obtained.

6.4 Initial Conditions for the Collection Network and Treatment Facilities

Initial conditions which are well chosen will assist the computer package in quickly finding a good final layout.

Initially the regional sewerage network consists of several sites where treatment plants may be located and many alternative routes (possible branching networks) which could transport the wastes to these potential treatment sites.

The initial conditions should allow each link an equal chance to be included in the first branching network which is determined. To ensure this, the following initial conditions were used.

All initial flow values in the links were set equal to a constant which is determined by the user. If different flow values were assigned to different links, then due to the nonlinearity of the cost curve (see Figure 3.2) the links with the higher flow values assigned to them would have lower cost coefficients (see Figure 6.5). These links in turn, would likely form the initial branching network.

The upstream elevation and diameter constraints due to incoming links were not considered. Thus, each link would have an upstream cover equal to the minimum cover.

The initial conditions are used in the calculation of the initial cost coefficient for each link. The cost coefficients, are in turn, based on the length of the link and the optimum diameter-excavation combination for gravity flow or the optimum force main-pump combination for nongravity flow. The optimum combination used for either type of flow is partially dependent on the topography for the link.

The initial flow value for each treatment facility is the smaller of the total wastewater flow in the network or the individual capacity of the treatment facility. This procedure will place no unecessary initial restriction as to the amount of waste which may be treated by an individual facility.

6.5 Selection of Flow Values, Upstream Elevations and Diameter Constraints for the Redunant Links

The magnitude of the flow variable returned from the linear programming algorithm will be finite for the links which comprise the branching network. The flow value will be zero if a link is not part of the branching network. Remembering that the treatment plants are represented in the linear programming algorithm by a slack variable then the value of the slack variable returned from the linear programming algorithm will be less than the predetermined capacity of a specific plant for the treatment plants which are part of the branching network. The magnitude of the slack variable for plants which are not part of the branching network will be equal to the capacity of each plant.

The links which are not, for any one iteration, part of the branching network are referred to as redundant links. In order to calculate a cost coefficient for each redundant link a finite flow value must be assigned to each link. The treatment plants which are not part of the branching network will not be referred to as redundant treatment plants since the flow variable which represents each of these plants will have a finite value. A plant flow and treatment efficiency will however, be required in order that a cost coefficient may be obtained.

There are several methods for choosing a flow value for a redundant link. These include:

1. Assign an arbitrary value of QLIMIT to all redundant links.
2. Assign a flow value to each redundant link which is equal to the average of all the nonzero flow values.
3. Assign a different flow value to each redundant link. This flow value would be equal to the total flow exiting from the upstream node of the branching link. This is, in fact, assigning a flow value to the redundant link which is equal to the flow value of the link in the branching network, of which the redundant link would be replacing.

That the choice of the flow value for the redundant link influences the value of the cost coefficient can be seen by examining Figures 6.4 and 6.5. Figure 6.4 and 6.5 show the cost vs. flowrate and cost coefficient vs. flow rate curves (cost equation given in Appendix B). If the flowrate for this link is assumed to be 1.00 then the cost coefficient would be equal to 5320/1.00 or 5320.0. If the flowrate is set equal to 5.00 then the cost coefficient

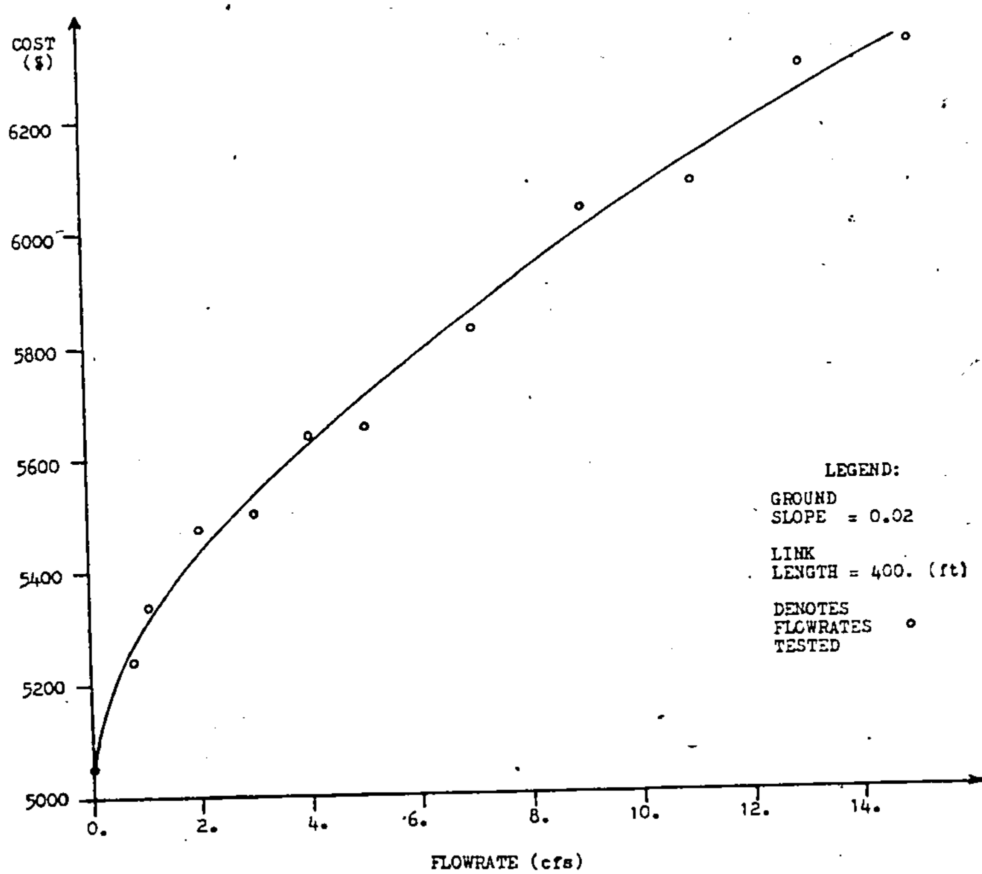


FIGURE 6-4 - COST vs. FLOW CURVE (TRANSPORTATION)

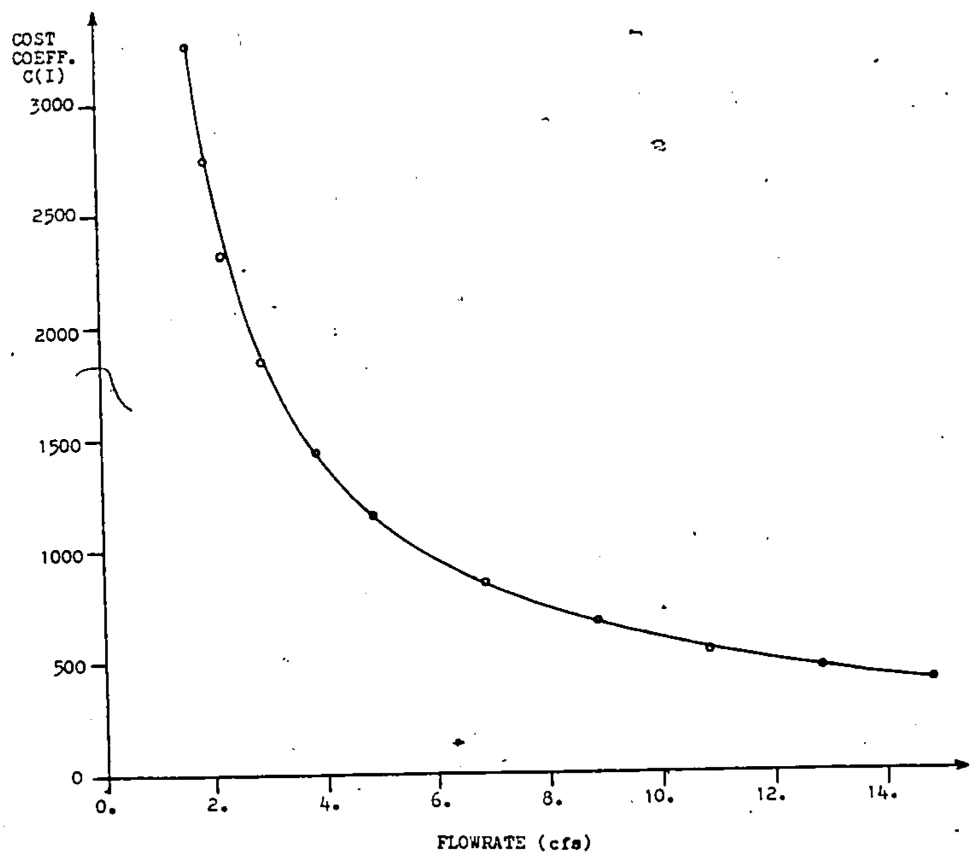


FIGURE 6·5 - COST COEFFICIENT vs. FLOW CURVE
(TRANSPORTATION)

would equal 5700/5.00 or 1140.0.

Subroutine SIMPLEX determines if a redundant link (nonbasic variable) should replace a member of the branching network (basic variable) by calculating a simplex coefficient for the redundant link.

The simplex coefficient for the nonbasic variable, Q_j, is given by the expression (see Chapter 4, section 1).

$$\Delta z_j = C_j - \sum_{i=1}^{NQ} A_{ij} \times C_i = C_j - z_j \qquad (4.2)$$

The nonbasic variable with the largest negative value of Δz_j will enter the solution. If all Δz_j values are positive then no improvement can be made to the solution and the current solution is considered to be optimal. The term $\sum_{i=1}^{NQ} A_{ij} \times C_i$ consists only of variables which are considered to be part of the basic solution and thus would not be affected by the flow value assigned to any redundant link.

Thus, it becomes clear that due to the degree of nonlinearity of the cost vs. flowrate curve that the values of flow assigned to each redundant link and therefore, the corresponding C_j value will largely determine whether or not Δz_j will be negative and thus whether or not Q_j will enter the basic feasible solution.

If the C_j values are too high the final solution found will be a poor solution since the nonbasic variables (redundant links) will have no chance to enter the basic feasible solution. The first branching network chosen will not change, and thus will taken to be the optimal solution.

If the C_j values are too low the program will cycle. Cycling is a phenomenon, whereby, in iteration i a nonbasic variable will replace a basic variable. In iteration (i+1) the basic variable which was replaced in iteration i will replace the new basic variable of iteration (i+1). This process will continue indefinitely.

Of the three methods which were previously mentioned, only the third method can ensure that reasonable flow values are assigned to each redundant link.

The first two methods, although computationally more efficient and easier to employ, will inevitably lead to one of the two problems mentioned above.

There are two other factors which will influence the value of the cost coefficients for the redundant links. They are, the upstream invert elevation constraint and the diameter progression constraint.

The upstream invert level of any redundant link i is set equal to the lowest downstream invert elevation of any incoming link from the branching network. A link in the branching network may have a flow in the same pipe, but in the opposite direction of the redundant link. This link would not be included when the upstream invert elevation for link i was determined (see Figure 6.6). If Q_4 was included in determining the upstream invert elevation for the redundant link with flow Q_3 then the cost coefficient for link 3 would be unreasonably high due to the excess excavation costs which would occur.

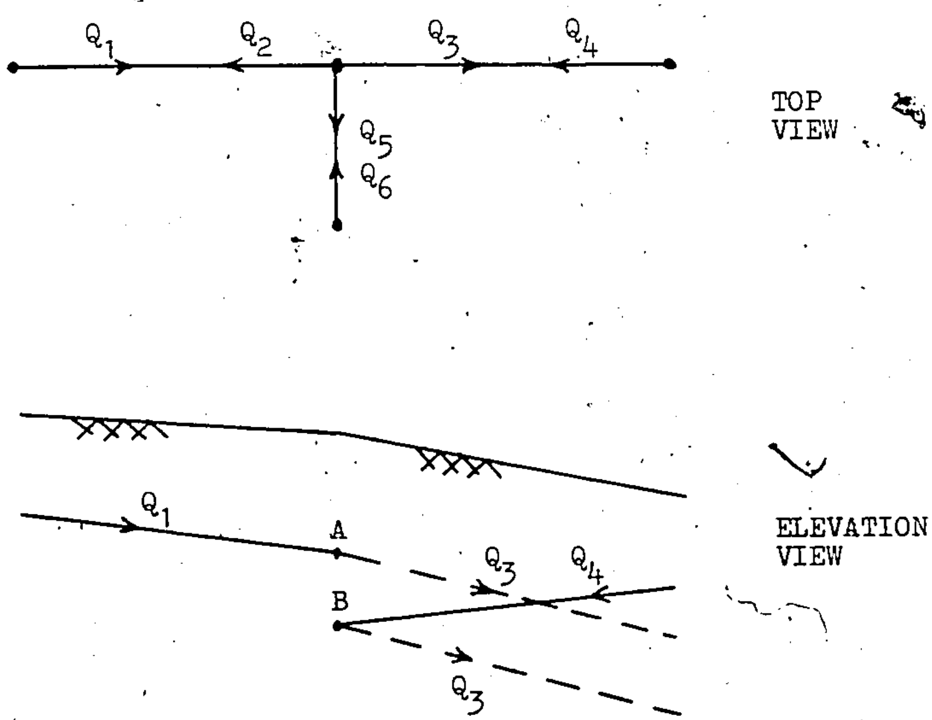
If there are no flows from the branching network entering the upstream node of a redundant link then only the minimum cover constraint would be taken into account.

The diameter progression constraint for a redundant link i will choose the largest incoming diameter from any incoming link in the branching network and use this diameter as the smallest allowable for link i. Again, if there is a link in the branching network which has a flow in the same pipe then this link is disregarded.

6.6 Selection of Flow Values and Efficiencies for Treatment Facilities which are not in the Branching Network

The cost coefficients for the treatment plants are based on the difference between the plant capacity and the plant flow (the slack variable). The cost coefficients are negative in sign.

Whereas there was a clear rationale for selecting the flow variable for links with zero flow the rationale is not as straightforward for treatment plants with zero plant flow. Treatment plants with zero plant flow will be part of



NOTES:

1) Q_1, Q_4 AND Q_5 ARE MEMBERS OF THE BRANCHING NETWORK.
Q_2, Q_3 AND Q_6 ARE REDUNDANT LINKS.

2) POINT A = U/S INVERT ELEVATION FOR LINK 3 IF ONLY Q_1 IS CONSIDERED WHEN DETERMINING THE U/S INVERT ELEVATION.

POINT B = U/S INVERT ELEVATION FOR LINK 3 IF BOTH Q_1 AND Q_4 ARE CONSIDERED WHEN DETERMINING THE U/S INVERT ELEVATION.

3) LINK I = LINK 3

FIGURE 6·6 - DETERMINATION OF THE UPSTREAM ELEVATION FOR A REDUNDANT LINK the basic solution since the slack variable will be finite.

In this study it was decided to calculate the cost coefficients for these plants so as the treatment plants would have a reasonable chance of being assigned a finite plant flow upon exit from the next linear programming iteration. Thus, the possibility of several different treatment plant configurations being tested within one computer run should be enhanced.

To do this the cost coefficients for plants with zero flow were calculated so as the C_j values were small in magnitude (but still negative in sign). This will tend to reduce the magnitude of the slack variable and thus increase the plant flow variable. This was facilitated by calculating the cost coefficient in the following manner.

1. Find the cost for the treatment plant when the plant is operating at capacity and the lowest efficiency allowable. Let RTN1 equal the plant cost and QCAP1 the plant flow.
2. Find the cost for the treatment plant when the plant is operating at one flow unit less than capacity and the lowest efficiency allowable. Let RTN2 equal the plant cost and QCAP2 the plant flow.
3. The cost coefficient is then equal to

$$C(SLACK) = -\frac{(RTN1 - RTN2)}{(QCAP1 - QCAP2)}$$

In conclusion, by restricting the redundant links to the same constraints as the links in the branching network and allowing the treatment plants with zero plant flow a reasonable chance to acquire a finite plant flow it will be ensured that the final solution found will be a good solution.

6.7 The Stopping Criteria

The method used to terminate the program is to compare the set of flow variables for iteration i to the set of flow variables for iteration (i+1). If any flow variable changes from iteration (i) to iteration (i+1) then this signifies that a basic variable was replaced by a nonbasic variable during the linear programming optimization. If there are no changes in any of the flow variables then the linear programming algorithm can find no further improvement and the current solution is assumed to be a least cost solution.

This method alone was not sufficient for terminating the program since cycling (see Chapter 6, section 5) generally occurred. The cycling generally involved 2 or at most 3 links which continually replaced each other from iteration i to iteration (i+1). To stop the program it was therefore necessary to select a maximum number of iterations which would be allowed before the program ended.

The number of iterations required to find the least cost solution is dependent on the nonlinearity of the cost functions. Typically, with the cost functions used in this study 5 to 7 iterations were required to find a least cost solution. If the cost functions are quite nonlinear then more iterations will be required. The user specifies the maximum number of iterations with the parameter ITER.

6.8 Incorporation of Pumps

When the terrain is steep the transportation network will, in all likelihood consist of gravitational flow only. Gravitational flow in a system where the terrain is gently sloping results in large excavation or pipe diameter costs. In this case installation of a pumping station may result in a lower overall cost. The tradeoff in cost is the additional cost required to install a pumping station as opposed to the excess excavation costs of a gravity system. The costs which are compared should not only be the estimated capital costs, but also the estimated running cost for the pumping station discounted to an equivalent present value. Appendix G details the costs included in calculating the total cost for a pumping station.

In Chapter 5, section 2, sub-section 5, it was explained how, for a branching network, subroutine SORT determined the order by which the cost for each link could be calculated. Basically, the procedure involved starting at the upstream ends of the network and working downstream.

The cost for link i could be determined only when the downstream invert elevations and sizes of pipes were known for all links upstream of link i.

The cost for a link with gravitational flow will be compared to the cost of a link with a pumping station for the furthermost upstream links first. A pumping station, if required, will be situated at the furthest upstream reach where the cost due to excess excavation exceeds the total pumping station costs. Since the cost of a pumping station is largely dependent on the flow this method will minimize the cost of the pumping station and should produce a good solution.

In any type of network the number of pumping stations in the final solution should be reduced. This reduces the possibility of system failures. The above method does not ensure that the final layout will have the minimum number of pumping stations, however, associated with the installation of a wastewater pumping station is an extremely large initial fixed cost. This large initial fixed cost should ensure that the number of pumping stations in the final solution is kept to a minimum.

Furthermore, an array called PENPMP has been added into the program package. This array allows the user to assign a cost penalty to a pumping station at any particular node. The array has several uses.

If the user feels that the location of a pumping station is undesirable due to aesthetic reasons then a large penalty term may be added to this pumping station. This will drive the pumping station out of the solution.

If the user feels that consolidating two pumps into one pump would produce a lower overall cost then large penalty terms should be applied to the two nodes where the existing pumping stations are.

A pumping station, may in certain instances produce an overall cost which is only marginally less than a solution consisting of gravitational flow only. Since a gravity flow system is more reliable it would be beneficial to compare the difference in costs. To ensure a gravity solution the user should apply a high cost penalty to all pumping stations.

CHAPTER 7

RESULTS AND CONCLUSIONS

7.1 Comparison of a Linear Programming Model to a Dynamic Programming Model for a Gravitational Collection Network

Any method, if it is to be useful in finding a good solution to the regional network problem must provide good results in the following two areas:

1. The method must choose a good layout out of the large number of layouts which have been selected by the engineer to be economically and technically feasible.
2. The method must, for the layout chosen determine a good depth-diameter combination for each link.

The dynamic programming method has the ability to account for a cost penalty further downstream due to small diameters and thus steep gradients being chosen upstream. For this reason, a dynamic programming model will, for a given layout produce a depth diameter combination for each link which will result in an overall minimum cost being found for the network.

Linear programming, on the other hand cannot account for the downstream cost penalties incurred due to the use of steep gradients upstream. Thus, for a given layout, it would not be expected that linear programming could obtain an optimal solution.

In order to determine the severity of not accounting for the downstream effects the circuited network of Fig. 7.1 was tested using both dynamic and linear programming.

In using the dynamic programming method, it was necessary for the circuited network to be broken down into a number of branching networks with known flow directions and magnitudes. The dynamic programming model used could not account for pumping stations, thus a gravity solution was found. No treatment plant costs were included since the system contains only a single location for a processing centre.

A dynamic programming model developed by Main [21] was used to calculate the cost for each branching network. This model uses a two pass system. In the first pass the program employs a wide range of permissable depths in relatively coarse increments of one foot to find the depth-diameter combinations resulting in the least cost.

The solution from this first pass is used to define a smaller range of depths to be searched in the second pass. The second pass searches this region in increments of 0.1 ft to find the optimal solution. The method results in depths which are accurate to ± 0.1 feet.

To apply the dynamic programming method, a total of eleven branching networks were identified as sub-sets of the

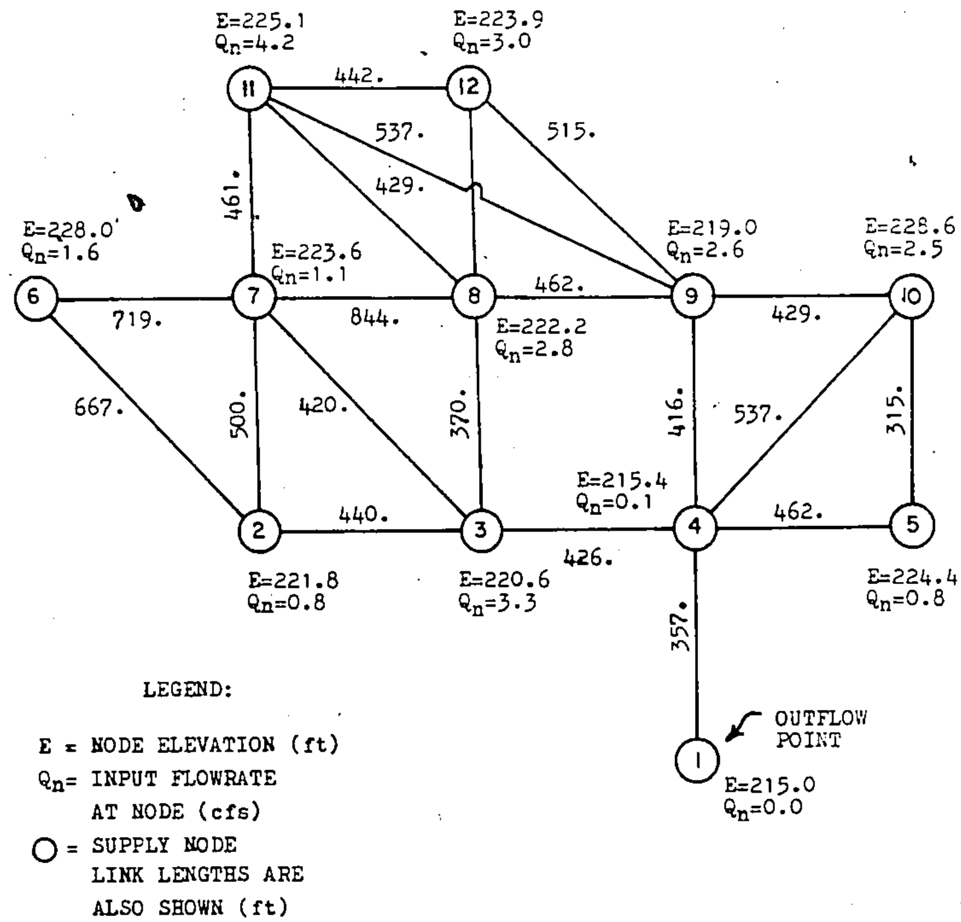
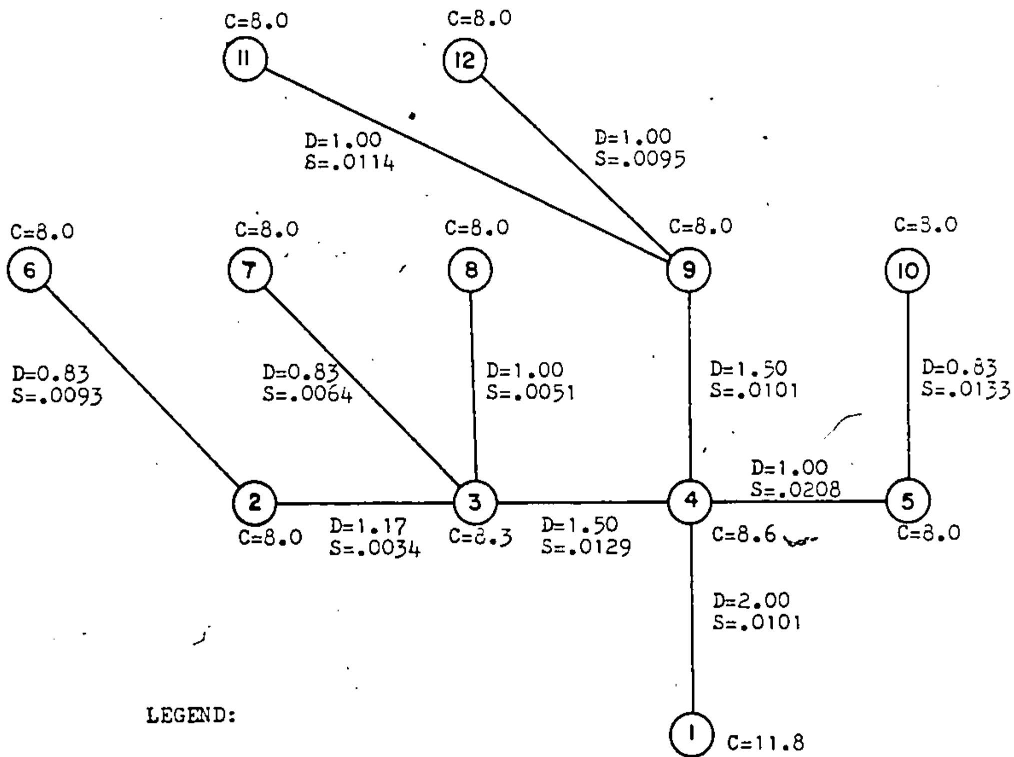


FIGURE 7·1 – SAMPLE 12 NODE CIRCUITED NETWORK USED FOR COMPARISON OF DYNAMIC AND LINEAR PROGRAMMING ALGORITHIMS original circuited network. Each branching sub-system was subjected to an analysis using Main's dynamic programming method. The network which resulted in the lowest cost, together with the corresponding depth-diameter combinations is shown in Figure 7.2. This was assumed to be the optimal solution against which other solutions could be compared. The other ten dynamic programming solutions obtained for the other branching networks had costs which exceeded the optimum by amounts ranging from 1.8% to 10.8%.

Using the circuited network, the linear programming method was employed in an attempt to find both the best layout and the corresponding depth-diameter combinations for each link. The solution initially obtained by linear programming is shown in Figure 7.3 and has a total cost in excess of the optimal one (Figure 7.2) by 4.2%.

As has been previously stated, there are two possible reasons as to why a solution found may not be optimal. The layout chosen may not be the best layout or, having chosen the optimal layout, the depth-diameter combinations chosen may not result in a least cost.

To find out which of the above reasons primarily resulted in a higher cost being found by the linear programming method the linear programming model was restricted to find the depth-diameter combinations for the same network in Figure 7.2. The results are shown in Figure 7.4. The



LEGEND:

C = GROUND COVER (ft)
D = PIPE DIAMETER (ft)
S = SLOPE
○ = SUPPLY NODE
NOTE: MINIMUM GROUND
COVER IS 8.0 ft

TOTAL COST = $72,870.

FIGURE 7·2- LEAST COST BRANCHING NETWORK
(DYNAMIC PROGRAMMING ALGORITHIM)

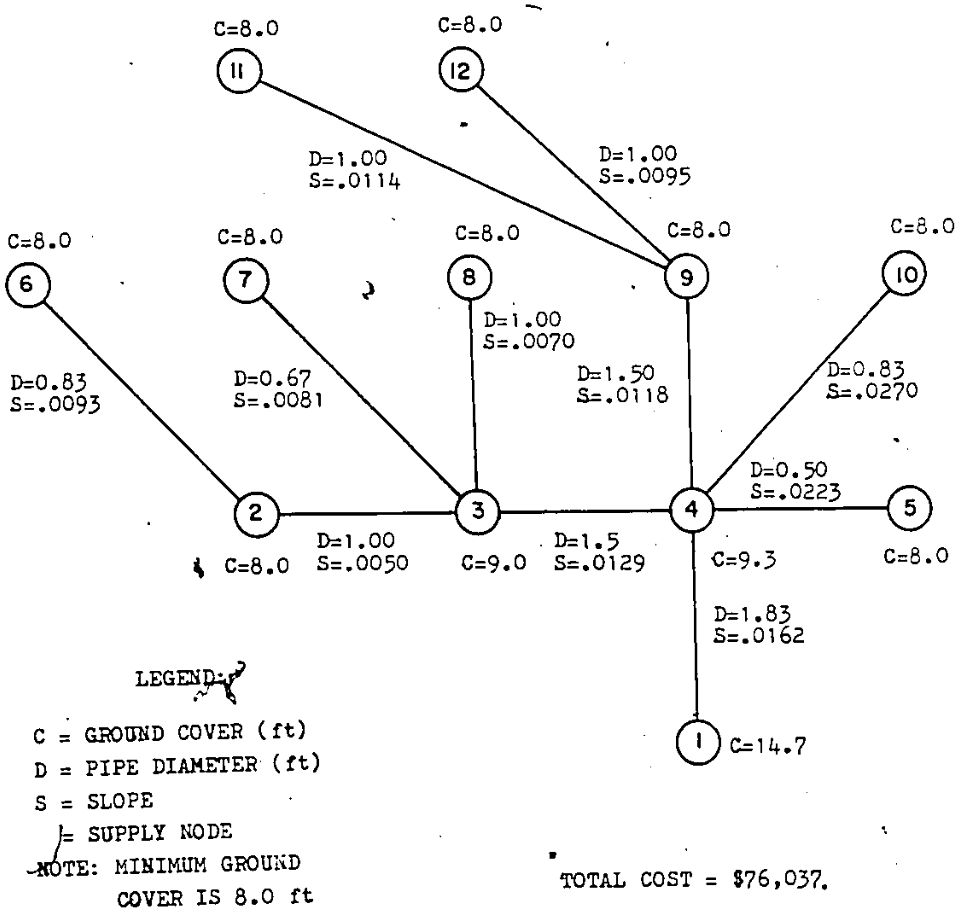
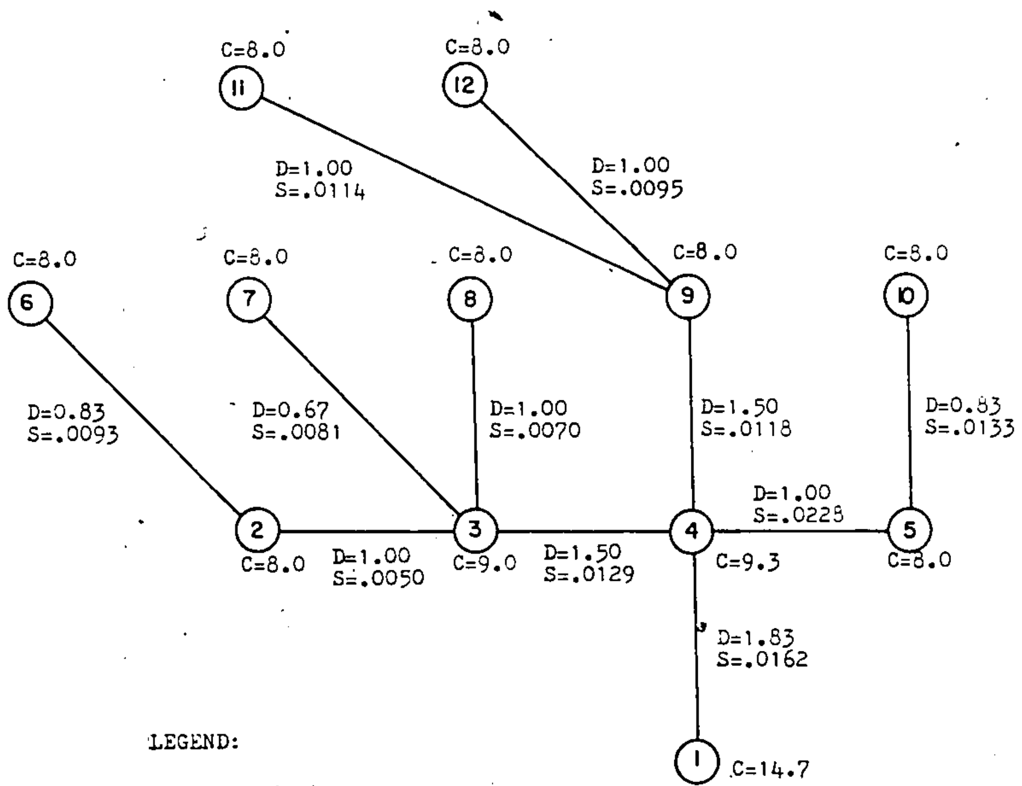


FIGURE 7·3 - LEAST COST BRANCHING NETWORK
(LINEAR PROGRAMMING ALGORITHIM)



LEGEND:

C = GROUND COVER (ft)
D = PIPE DIAMETER (ft)
S = SLOPE
= SUPPLY NODE
NOTE: MINIMUM GROUND COVER IS 8.0 ft

TOTAL COST = $73,402.

FIGURE 7·4 - LEAST COST SOLUTION FOR A SPECIFIC NETWORK (LINEAR PROGRAMMING ALGORITHIM) difference in cost found by the two methods is 0.7%.

For this example, the major reason for the difference in cost between the two methods is that the linear programming method did not choose the best layout. Comparison of Figure 7.2 and 7.3 shows that the networks differ by only one link. The reason for this difference can be explained as follows. The SIMPLEX algorithm will be used to illustrate the problem.

Each supply node is initially represented in the basic feasible solution by an artificial variable with a high cost coefficient. The high cost coefficient assigned to these variables ensures that these variables are the first to leave the basic feasible solution.

For node #10, the flow variable replacing the artificial variable would be one of Q_{23}, Q_{30} or Q_{40} (see Figure 7.5). The flow variable Q_{40} has the largest negative simplex coefficient and therefore this variable enters the basic feasible solution.

The values of the first set of simplex coefficients are dependent on the initial conditions assigned to the problem. Therefore it is important that these initial conditions be realistically defined. The initial conditions are described in Chapter 6, section 4, and for the reasons stated in that chapter these initial conditions should not prevent the linear programming algorithm from

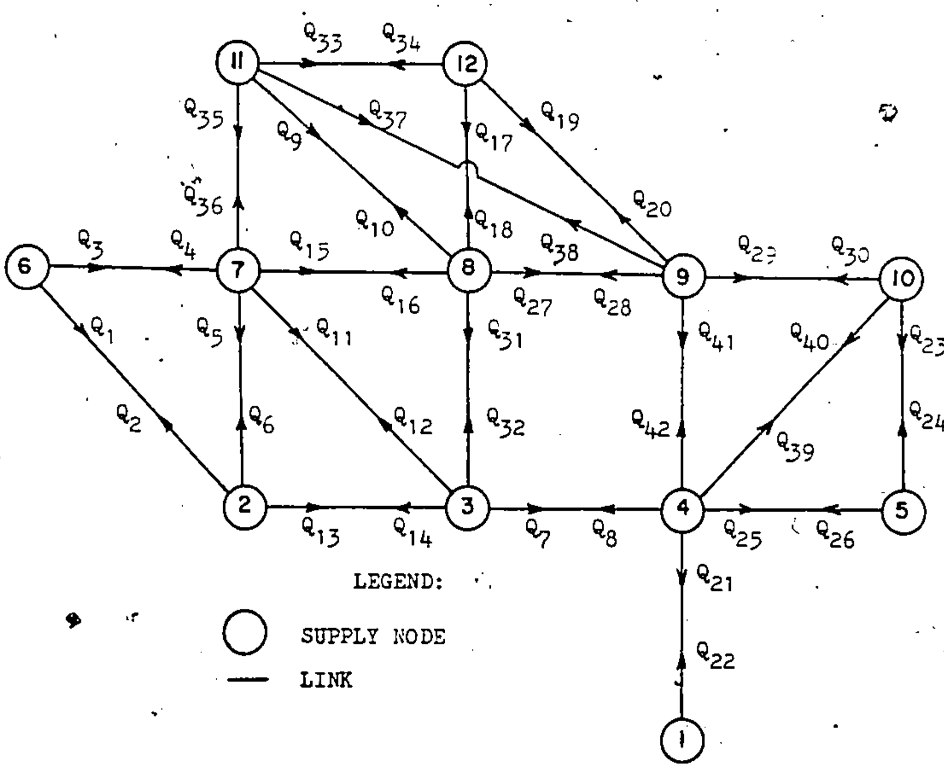


FIGURE 7·5 - SAMPLE 12 NODE CIRCUITED NETWORK INDICATING FLOW VARIABLES finding the network leading to a least cost solution.

The first basic feasible solution chosen by SIMPLEX includes the variable Q_{40}. The cost coefficients are then recalculated for all links using the updated flows.

Subroutine SIMPLEX is entered a second time and a check is made to see if the simplex coefficients for any nonbasic variables are negative. Since there are no negative simplex coefficients the branching network remains unchanged and is assumed to be optimal.

A detailed account of how the linear programming algorithm is used in conjunction with a number of subroutines to find the minimum cost for a specific branching network is detailed in Chapter 5. Briefly, the procedure is as follows; for a given flow, the diameter is found which results in a least cost for each link. For each link there are a set of technical constraints that must be satisfied. The total cost for the network is then the summation of the individiual minimum costs for each link.

If a dynamic programming method is used then the cost penalties downstream due to a steep gradient being chosen upstream will be accounted for. Thus, the diameter chosen for a specific link may not lead to a minimum cost for that link. It will however, result in an overall minimum cost for the network.

Therefore, for a given layout the primary reason for the difference in total cost between the two methods is that a linear programming method cannot account for the cost penalites downstream due to steep gradients being employed upstream.

The difference in the minimum total cost found by each method for the network used in this comparison (see Figures 7.3 and 7.4) was 0.7%. The magnitude of this difference is dependent on several factors which will subsequently be discussed.

When the ground surface is steep the excavation depth for each link is likely to be a minimum, provided the maximum velocity constraint does not become binding. In this situation the cost penalities are likely to be negligible and the depth-diameter combinations found by a linear programming method will be quite similar to those found by a dynamic programming method.

As the surface slope becomes milder there is increasing dependence on the ratio of the pipe costs to the excavation costs. The excavation depths are likely to increase and the opportunity for cost penalties downstream due a steep gradient upstream increases. However closer inspection of the problem and the associated technical constraints will show that even if the surface slope is mild that the cost penalties incurred are not likely to be large.

Walsh and Brown [34] stated that typically the excavation and backfill costs may account for as much as 80% of the total cost of the sewer. This percentage will increase as the average excavation depth increases. Generally the lengths of the reaches under consideration are quite long. For a reach with a mild slope the selection of a small diameter, and thus a steep gradient would result in a large excavation depth at the downstream end of the link. This in turn, would result in a high cost for this link. Thus, there is a tendency for large diameters (shallow gradients) producing the minimum cost in reaches with mild surface slopes.

The incorporation of the technical constraints and the minimum and maximum design flows greatly restricts the number of diameters which may be used in any one reach. Typically the maximum flow at the end of the design period will be four to ten times as large as the minimum flow at the beginning of the design period. Any diameter selected must be able to satisfy all the technical constraints such as minimum and maximum velocities and minimum cover while carrying both of these design flows. Smaller diameters, although able to carry the minimum design flow and satisfy the minimum velocity constraint are not likely to have the capacity to handle the maximum design flow and at the same time satisfy the maximum velocity constraint. This further reduces the chance of large penalty costs being incurred due to excessively small diameters being selected upstream.

Therefore, for a specific branching network the depth-diameter combinations and the minimum cost found using the linear programming algorithm and associated subroutines should, under most conditions produce results which are similar to those found using a dynamic programming method. However, in regions where the ground surface is mild over a long distance ignoring the downstream effects due to choosing steep gradients upstream could result in excessive excavation depths in the downstream links and thus a significantly higher overall cost. For this reason the program allows for the installation of a pumping station at the upstream end of any reach. The excavation depth along the entire length of the reach will thus be reduced to the minimum amount allowed. A pumping station will be installed when the total cost of a link including the pumping station is less than the least cost of the gravity sewer for that link.

It is worthwhile to point out that several authors who have attempted to find the best depth-diameter combinations for a given layout did not include the possibility of pumping in their programs. Argaman, et al. [3] who used a dynamic programming method and Dajani, et al. [8] who used a linear programming method both selected pipe elevation as the design variable. The inclusion of pumping into these programs would therefore be quite difficult.

The previous discussion has been limited to the question of selecting the depth-diameter combinations for a given layout. If the inclusion of all possible alternative routes is considered then the likelihood of a link which causes large cost penalties downstream due to its steep gradient being included in the final layout is further reduced. The reason for this is that all the links downstream of this link will have high costs due to the large excavation depths. This set of depth-diameter combinations will not result in an overall minimum cost.

If an alternative link with a shallow gradient replaces the link with the steep gradient then the costs of the downstream links are likely to reduce since excavation costs will reduce. This set of depth-diameter combinations will result in a lower overall cost.

In conclusion, this comparison shows that this linear programming algorithm and associated subroutines do indeed find, from a large number of alternative layouts both a good layout and depth-diameter combinations for each link in this layout. Eleven branching networks were analyzed using the dynamic programming method. It was thought that any one of these branching networks could have led to a minimum cost. The cost found by the linear programming method (Figure 7.3) was as low as all but two of the costs found using dynamic programming.

In addition, a large difference in the costs was due to the wrong layout being chosen by the linear programming method. This type of solution is called a local optima. Local optima occur because the nonlinear concave objective function is being minimized subject to linear constraints (see Appendix C).

Tufgar [33] presented a subroutine called HYVRST which may be used in conjunction with a linear programming algorithm. This method does ensure that the global optima would be chosen, however, it was found by Tufgar to be computationally expensive.

7.2 A 40-Node Circuited Network

The final least cost solution involves the selection of the transportation and processing network which leads to the fullest utilization of the assimilative capacity of the river.

As was explained in Chapter 4, section 3, there are a very large number of sets of treatment plant flows. Each set of treatment plant flows will lead to varying degrees of utlization of the assimilative capacity of the river and thus varying plant costs.

In addition, if the water quality constraints are totally relaxed then the least cost for this type of problem may be found by finding the best combination, from the many possible combinations available, of the transportation routes and the associated treatment plant sizes and locations.

The problem then becomes one of combining these two types of problems such that the additional transportation and processing costs which result are minimized. The network which minimizes these additional costs is referred to as the resultant least cost network or lowest cost network.

The computer program uses an iterative technique in an attempt to find the resultant least cost network. For a given set of cost coefficients, C, the linear programming algorithm will yield an improved estimate of flows, Q'. A new set of cost coefficients will then be used to yield another set of flow values. If the resultant least cost is to be found then one of the sets of flow values will be such that the fullest assimilative capacity of the river will be utilized. Whether this will occur depends on the ability of the computer program's iterative procedure to converge to this best set of flows, Q*.

The circuited network of Figure 7.6 was one of the circuited networks which was tested in order to determine the program's effectiveness. There are 40 nodes in the

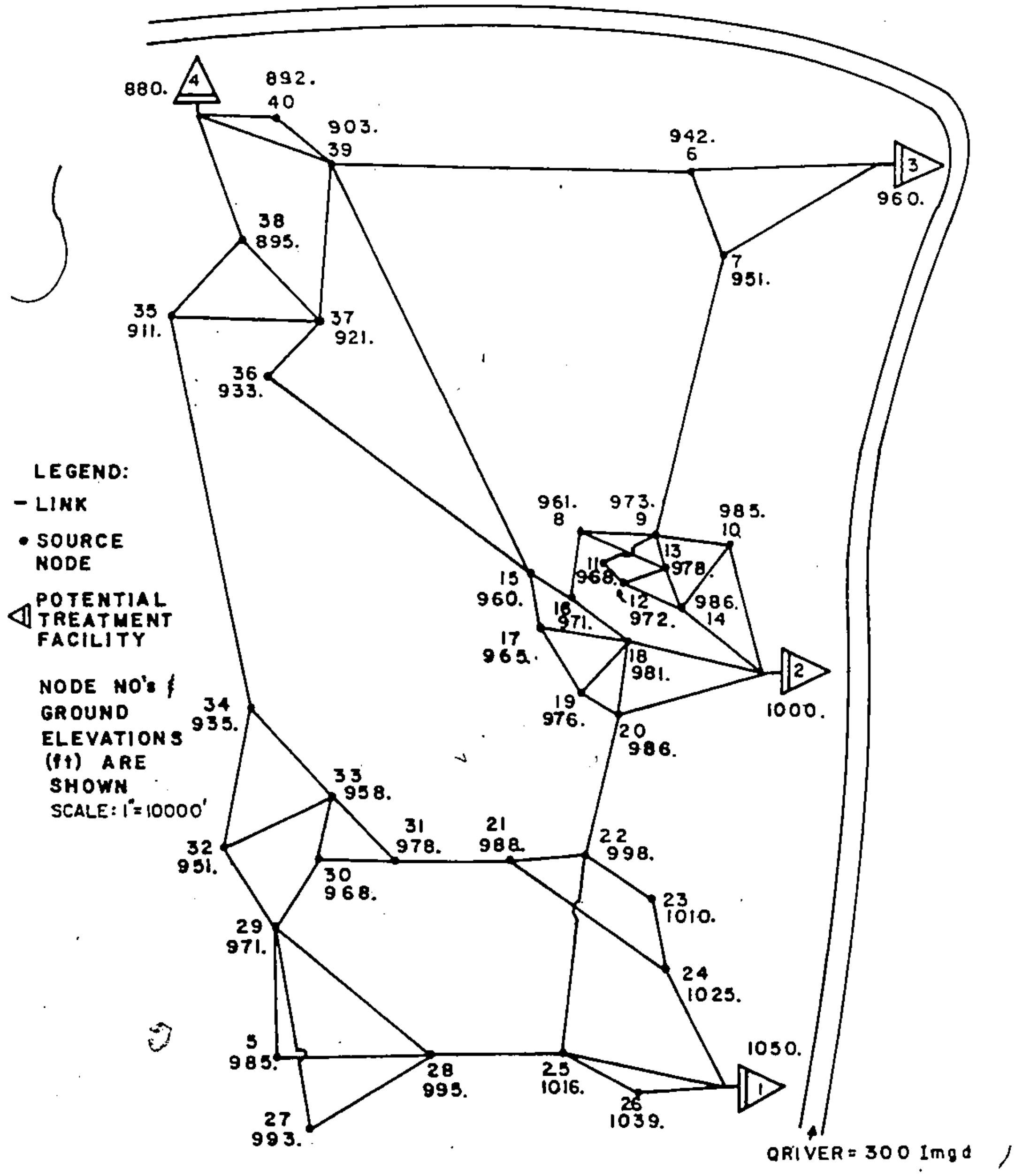


FIGURE 7·6 (a) - SAMPLE 40 NODE CIRCUITED NETWORK
(NODE NUMBERS AND GROUND ELEVATIONS)

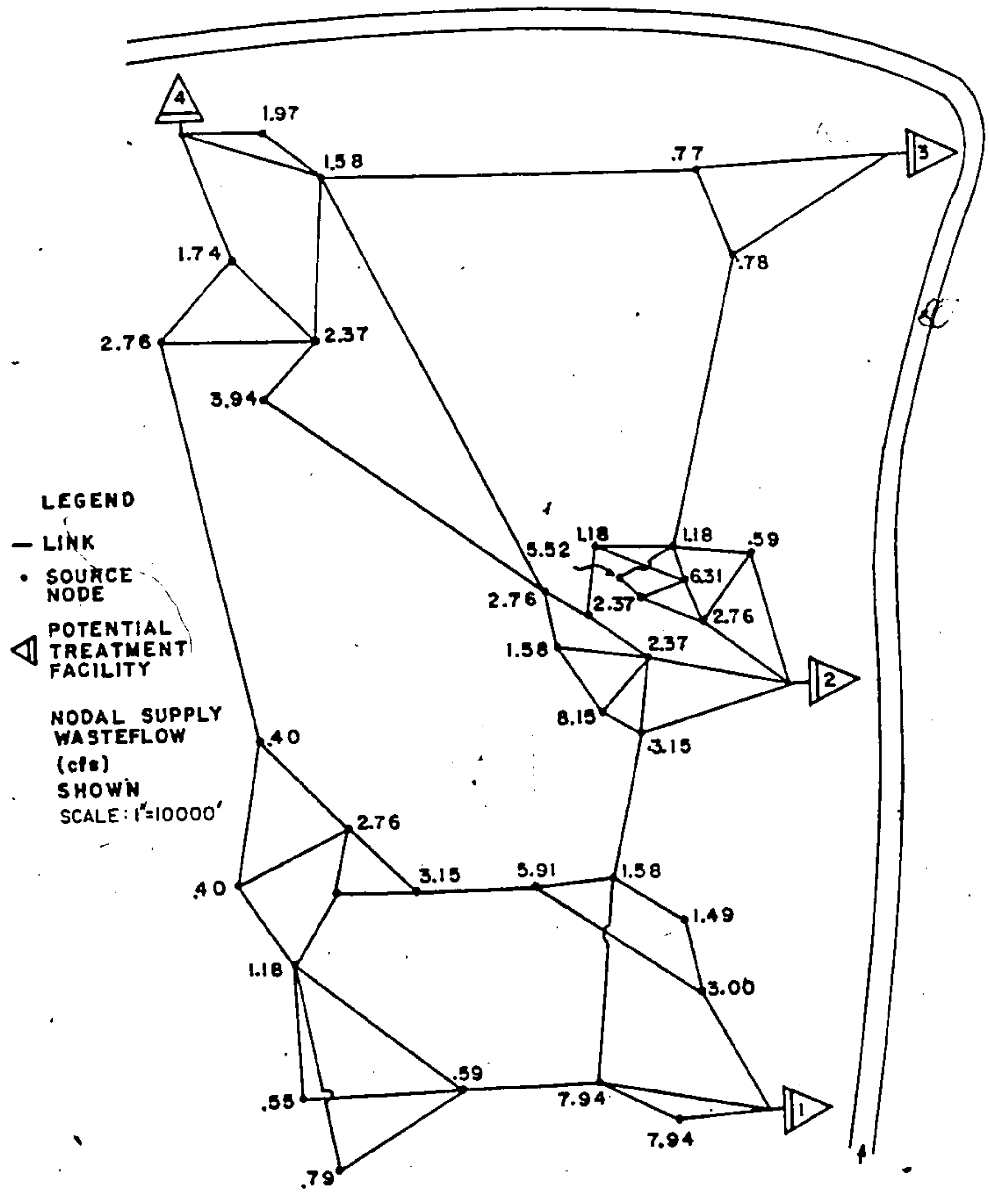


FIGURE 7·6 (b) - SAMPLE 40 NODE CIRCUITED NETWORK (NODAL SUPPLY WASTEFLOWS)

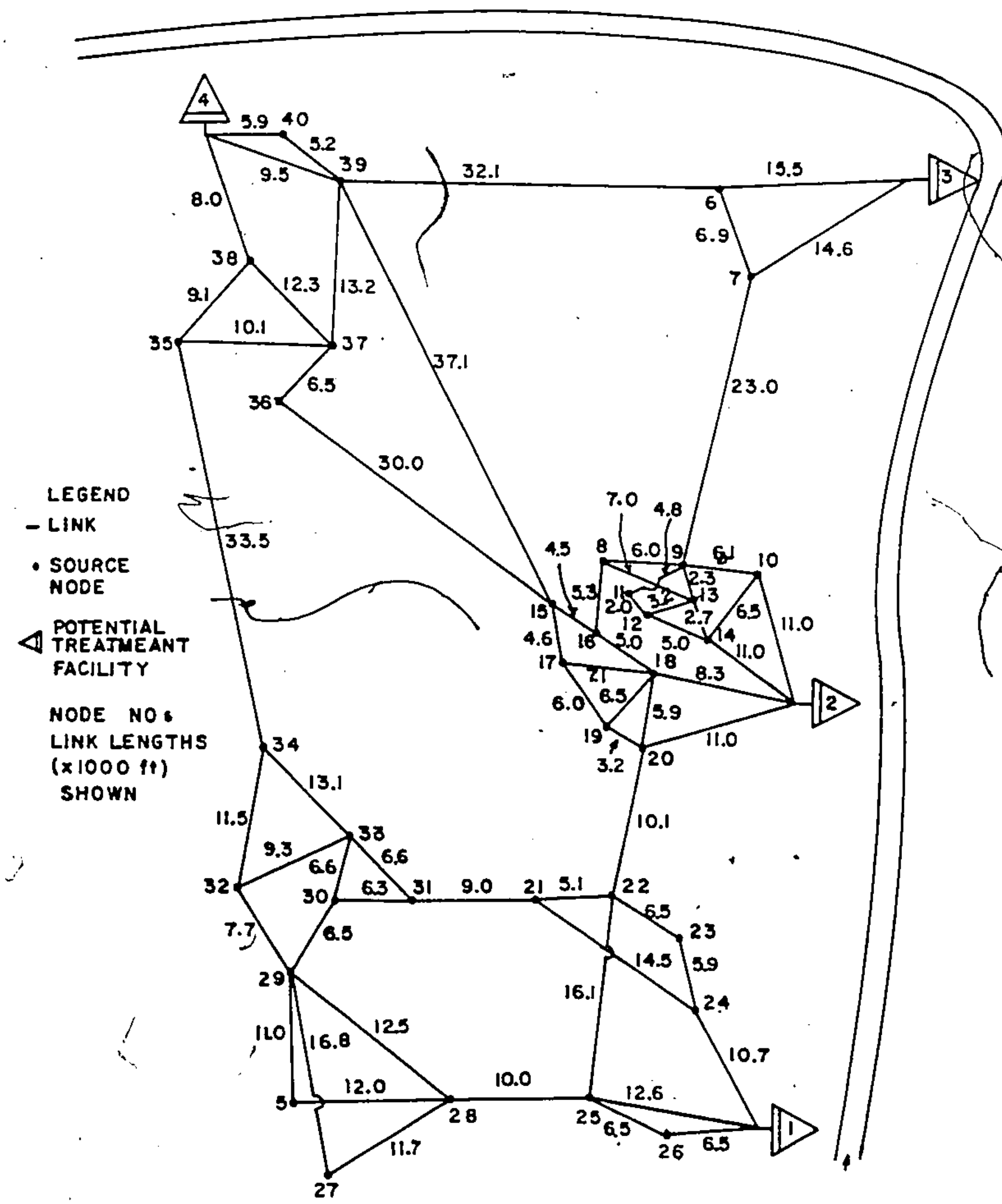


FIGURE 7·6 (c) - SAMPLE 40 NODE CIRCUITED NETWORK (NODE NUMBERS AND LINK LENGTHS) problem, of which 36 are supply nodes and 4 are potential treatment facilities. There are 66 feasible links (with flow being allowed in either direction for all links). The total wastewater flow from all the supply nodes is 93.88 cfs. It was assumed that each potential treatment plant had the available capacity to handle the entire wastewater if necessary.

The maximum allowable oxygen deficit was calculated to be 4.00 mg/l. This is based on a saturation oxygen concentration of 9.0 mg/l (at 19°C) and a minimum allowable dissolved oxygen concentration of 5.0 mg/l. Fifty discrete levels of oxygen deficit were possible so that the final water quality will be accurate to $\pm$0.04 mg/l. The initial streamflow selected was 300 Imperial million gallons per day (Imgd) or 161.6 cfs. The flow value selected is commonly based on the 7 day - 10 year low flow for the river. Other data which is required for the treatment plants and river reaches is given in Table 7.1. Table 7.2 lists relevant data which is common for the links.

The least cost branching network which satisfies the water quality standard is shown in Figure 7.7*. The total

* The final solution contains a large number of force main pumping station schemes. This is due to an inconsistency in the cost equations which were used to determine the force main costs and the gravity sewer costs. The costs used were such that at large flows (and therefore large diameters) a force main would be considerably cheaper than a gravity sewer (for the same diameter). Appendix G lists the cost equations which were used to determine the costs for the force mains and the gravity sewers.

TABLE 7·1 - DATA FOR TREATMENT PLANTS AND RIVER REACHES

Parameter	Treatment Plant #1	Treatment Plant #2	Treatment Plant #3	Treatment Plant #4
Rearation Coefficient*	0.23	0.19	0.25	0.28
Deoxygenation Coefficient*	0.17	0.14	0.15	0.13
Dissolved Oxygen Deficit of Wastewater (mg/l)	8.0	7.0	7.5	6.5
BOD Concentration at inlet of Plant (mg/l)	350.	350.	350.	350.
Time of travel for reach (days)	0.37	0.21	0.09	1.50

* based on T = 19^{o} for this example.

NOTES:

The following treatment efficiencies were possible for all treatment plants (% BOD removal) 35,45,55,60,65,70,75,80,84,86,87,88,89,90,91,92,93,94,95,96.

The levels of oxygen deficit were discrete values from 0.00 mg/l to 4.00 mg/l in increments of 0.08 mg/l.

Minimum Cover	= 8.0 ft.
Minimum Pipe Diameter	= 8.0 in (.667 ft)
Infiltration	= 0.00186 (ft^3/acre/sec)

Permissable pipe sizes (ft)

0.667, 0.833, 1.00, 1.25, 1.50, 1.75, 2.00, 2.25, 2.50, 2.75, 3.00, 3.50, 4.00, 6.00

TABLE 7·2 - DATA FOR TRANSPORTATION LINKS

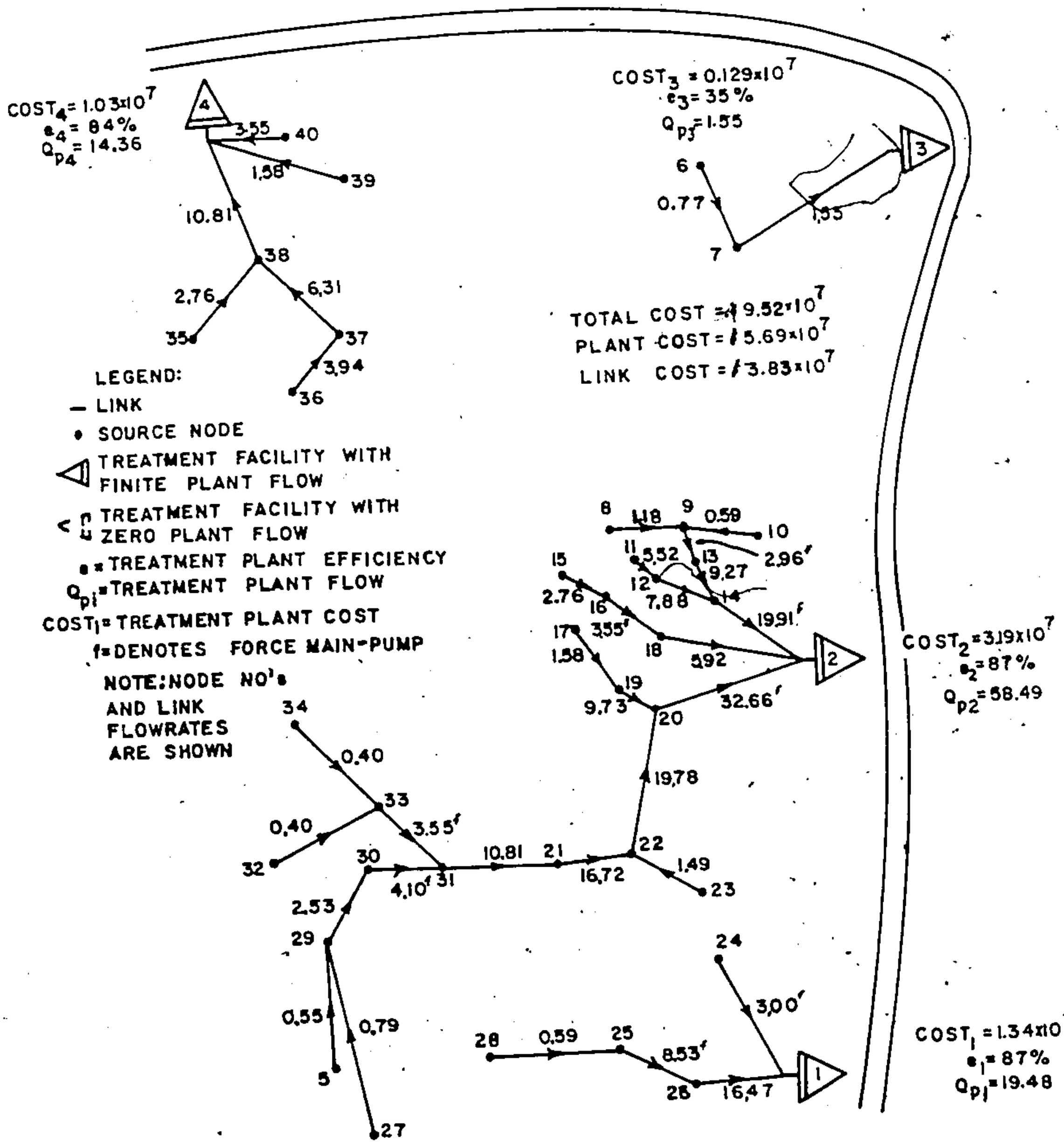


FIGURE 7·7- INITIAL LEAST COST BRANCHING NETWORK least cost was found to be $95,200,000 of which $56,900,000 represented treatment plant costs and 38,300,000 was for link costs. The least cost was found in 5 iterations.

Table 7.3 shows the individual plant flows, efficiencies and cost coefficients after each iteration. The cost breakdown for each iteration is also given. The total plant costs and total link costs for iteration 1 are high since initially all links are assumed to be part of the network and all treatment facilities are assumed to handle the total flow.

Analysis of the treatment plant costs from iteration 2 to iteration 5 indicates that the initial (iteration 2) set of treatment plant flows selected have resulted in good utilization of the assimilative capacity of the river. This is reflected by the fact that the total plant cost was reduced only slightly from iteration 2 to iteration 5. Since the plant flows changed only slightly through iterations 2 to 5 this would imply that the cost reduction is occurring within the transportation part of the network only. Therefore, the selection of the initial (iteration 2) transportation route or the depth-diameter combinations for the route selected must not have been a good one.

In the previous section it was proven that if the objective function consisted only of costs due to transporting the wastewater then a good final branching network could

Plant Number (cfs)	Plant Flow (%)	Plant Eff. ($\times 10^7$)	Plant Cost	Plant Cost Coeff.	
					ITERATION #1
1	93.89	92	5.09	-319447	Total Cost = \$264.9 x 10^7
2	93.89	90	4.91	-306972	Plant Cost = \$16.58 x 10^7
3	93.98	35	3.29	-205854	Link Cost = \$248.32 x 10^7
4	93.89	35	3.29	-205854	
					ITERATION #2
1	23.49	87	1.55	-325180	Total Cost = \$11.74 x 10^7
2	53.68	87	2.98	-305683	Plant Cost = \$5.76 x 10^7
3	1.55	35	0.13	-254145	Link Cost = \$5.98 x 0^7
4	15.15	87	1.10	-333491	
					ITERATION #3
1	20.27	84	1.36	-322652	Total Cost = \$10.67 x 10^7
2	57.70	87	3.16	-303748	Plant Cost = \$5.76 x 10^7
3	1.55	35	0.13	-254145	Link Cost = \$4.91 x 10^7
4	14.36	90	1.11	-354240	
					ITERATION #4
1	20.27	84	1.36	-322625	Total Cost = \$9.89 x 10^7
2	57.70	8[illegible]	3.16	-303748	Plant Cost = 5.76 x 10^7
3	1.55	3[illegible]	0.13	-254145	Link Cost = \$4.13 x 10^7
4	14.36	9[illegible]	1.11	-354240	

TABLE 7·3 - SUMMARY FOR INITIAL LEAST COST NETWORK

Plant Number (cfs)	Plant Flow (%)	Plant Eff.	Plant Cost ($\times 10^7$)	Plant Cost Coeff.
1	19.48	87	1.34	-328903
2	58.49	87	3.19	-303379
3	1.55	35	0.13	-254145
4	14.36	84	1.03	-328826

ITERATION #5 :

Total Cost = $\$9.52 \times 10^7$
Plant Cost = $\$5.69 \times 10^7$
Link Cost = $\$3.83 \times 10^7$

be found. The fact that the transportation costs decreased significantly without a corresponding increase in the processing costs (for a set of treatment plant flows which varied by only a small amount) would suggest that the inclusion of the processing costs and a water quality standard into the objective function still results in a good transportation route being selected as part of the final branching network.

In order to find out how good this least cost solution is a series of 7 different computer runs were carried out.

The computer program has been developed so that the user may place certain types of restrictions on the circuited network. The restrictions may either eliminate certain possible branching networks from being involved in the final solution or force a certain branching network to comprise the final solution.

One type of restriction is to reduce or eliminate the capacity at certain potential treatment plants. An extreme example of this type of restriction would be to eliminate all of the capacity for treatment plants 1 through 3 in Figure 7.6, thereby forcing all the supply flow to treatment plant 4. Another more moderate example would be to reduce the capacity at plant 4 to say 30.0 cfs. The maximum supply flow to this plant would then be reduced to 30.0 cfs. The individual plant capacities may easily be modified by changing the appropriate element in the BMAX matrix.

Another type of restriction is to place a large penalty cost on certain links in the circuited network. This will have the effect of removing these links from the solution and thus the flow will be directed through alternative links. The penalty costs may be applied to any link through the use of the array CPEN.

Through good selection of the restrictions it should be virtually ensured that the least cost from one of the computer runs will result in the resultant least cost solution.

The 7 least costs from each computer run as well as the least cost from the initial run (referred to as run 0) are listed in Table 7.4. The table shows that run 4 resulted in the lowest cost.

The restriction placed on the initial circuit network for run 4 was that the capacity at treatment plant 1 was reduced to zero. Figure 7.8 illustrates the branching network which resulted in the lowest cost. Table 7.5 lists the individual plant flows, efficiencies and cost coefficients for each iteration for run 4.

It was previously mentioned that the resultant least cost network is the network which minimizes the additional transportation and processing costs due to the combining of

Run No.	Total Cost x $\$10^6$	% Higher than Lowest Cost
2	102.4	10.6
7	102.3	10.5
1	97.8	5.6
3	97.6	5.4
5	97.6	5.4
6	96.3	4.0
0	95.2	2.8
4	92.6	0.0

TABLE 7·4 - SUMMARY OF LEAST COSTS

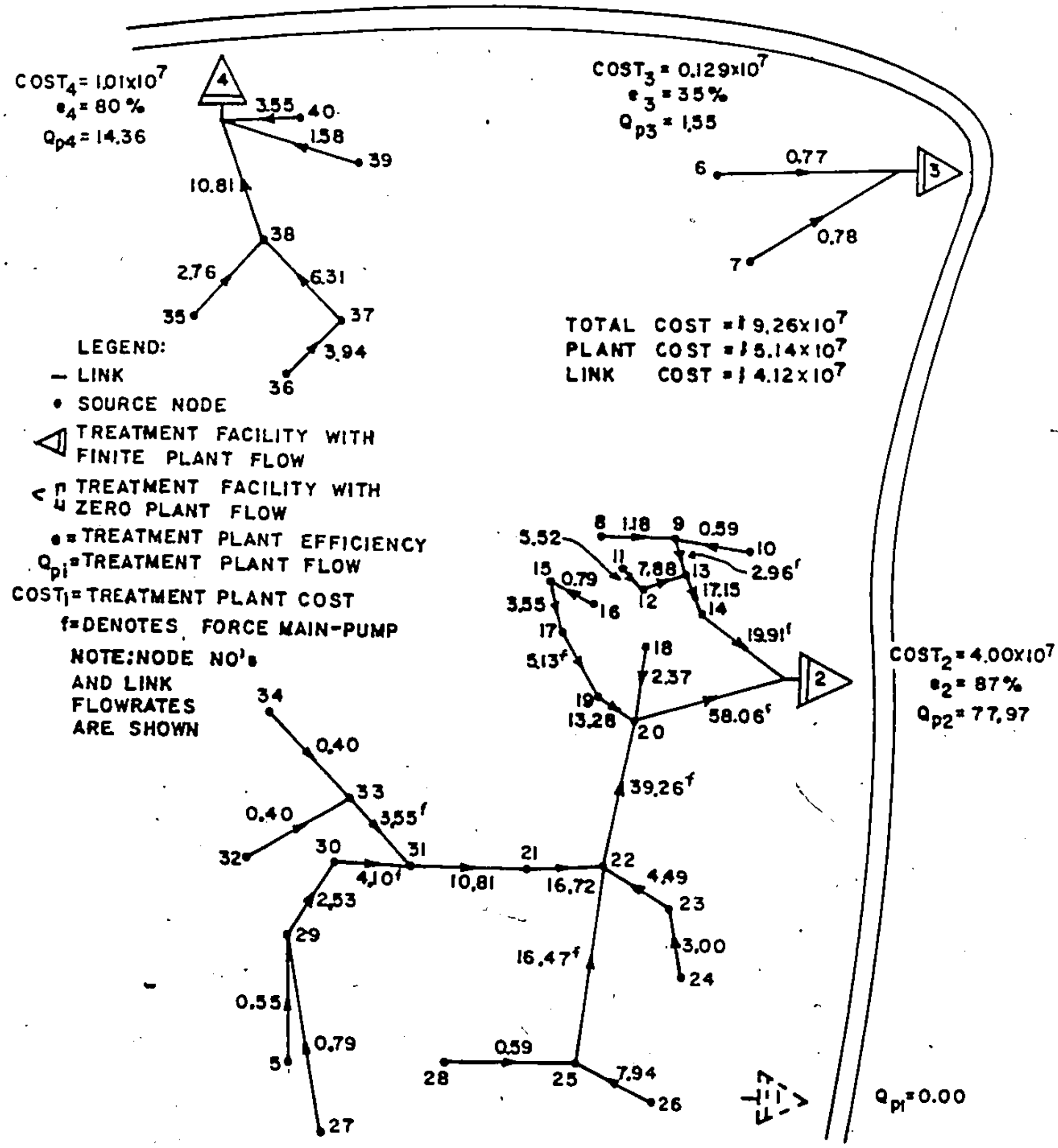


FIGURE 7.8 - RESULTANT LEAST COST BRANCHING NETWORK

Plant Number (ofs)	Plant Flow (%)	Plant Eff. (x10^7)	Plant Cost	Plant Cost Coeff.	
					ITERATION #1
1	93.89	0	0	0	Total Cost = \$272.0 x 10^7
2	93.89	99	11.03	-742700	Plant Cost = \$23.7 x 10^7
3	93.89	96	7.58	-510800	Link Cost = \$248.3 x 10^7
4	93.89	92	5.10	-318447	
					ITERATION #2
1	0	0	0	0	Total Cost = \$9.63 x 10^7
2	60.25	88	3.33	-308525	Plant Cost = \$5.41 x 10^7
3	1.55	35	0.13	-254145	Link Cost = \$4.22 x 10^7
4	32.07	84	1.95	-313102	
					ITERATION #3
1	0	0	0	0	Total Cost = \$9.33 x 10^7
2	74.65	87	3.87	-296554	Plant Cost = \$5.22 x 10^7
3	1.55	35	0.13	-254145	Link Cost = \$4.15 x 10^7
4	17.68	84	1.22	-325199	
					ITERATION #4
1	0	0	0	0	Total Cost = \$9.29 x 10^7
2	77.97	87	4.00	-295296	Plant Cost = \$5.14 x 10^7
3	1.55	35	0.13	-254145	Link Cost = \$4.15 x 10^7
4	14.36	80	1.01	-321375	

TABLE 7.5 - SUMMARY FOR RESULTANT LEAST COST NETWORK

Plant Number (cfs)	Plant Flow (%)	Plant Eff. $(\times 10^7)$	Plant Cost	Plant Cost Coeff.	
					ITERATION #5
1	0	0	0	0	Total Cost = $9.29 x 10^7
2	77.97	87	4.00	-295296	Plant Cost = $5.14 x 10^7
3	1.55	35	0.13	-254145	Link Cost = $4.15 x 10^7
4	14.36	80	1.01	-321375	
					ITERATION #6
1	0	0	0	0	Total Cost = $9.26 x 10^7
2	77.97	87	4.00	-295296	Plant Cost = $5.14 x 10^7
3	1.55	35	0.13	-254145	Link Cost = $4.12 x 10^7
4	14.36	80	1.01	-321375	

the two separate types of problems. If the processing and transportation costs for the branching networks of Fig. 7.7 and Fig. 7.8 are compared then the following will be found.

The total transportation cost for the branching network of Figure 7.7 is $2,900,000 less than that of Fig. 7.8. The plant costs, however, are more expensive by $5,500,000. It may therefore be stated that the total additional costs for the branching network of Figure 7.8 are less than those of Figure 7.7 by $2,600,000.

There are two major reasons for the lower cost of the branching network in Figure 7.8. Both of these reasons indicate that the assimilative capacity of the river has been more fully utilized by this branching network. The first reason that a lower cost has been attained is due to the lower efficiency at plant 4 (the flow value at plant 4 is the same for both branching networks). The second reason is that the economies of scale inherent in building one large plant (plant 2) outweigh the additional transportation costs of the wastewater to plant 2 (it should be noted that the efficiency at plant 2 remained constant. An increase in the treatment efficiency may offset the savings due to the economies of scale).

7.3 Conclusions

The results found for the circuited network of Figure 7.7 were consistent with results found for other circuited networks that were tested. The iterative linear programming procedure would not, in general converge to the resultant least cost network. The primary reason that the resultant least cost network was not found is due to the program's inability to converge to the set of plant flows which most fully utilize the assimilative capacity of the river.

The initial solution found did however result in a set of flows which resulted in good utilization of the assimilative capacity of the receiving body of water and a final cost which was only 2.8 percent higher than the lowest cost found.

The program has been developed in such a manner that the procedure which was used above to find the lowest cost could easily be used to find the lowest cost for any circuited network of a similar size to the one shown in Figure 7.6. Once the initial least cost branching network has been attained the planner, by combining intuition and common sense approaches with the two types of restrictions which may be placed on the circuited network, may easily generate alternative least cost solutions. By using this procedure (which typically requires 6 to 8 computer runs) the planner should be assured of a lowest cost solution.

The computational cost required to find this solution will be small. Typically, one computer run with 10 iterations required 5.0 seconds of compilation time and 105 seconds on the CDC6400 computer.

It should also be kept in mind that the lowest cost found is optimal from an economic standpoint only (although the desired dissolved oxygen standard will be satisfied). For political or other reasons the economic optimum may well not result in the overall best solution. Thus, this method of generating alternative solutions will allow the planner to address the alternative plans and compare them to the economic optimal.

7.4 Summary

The objective of this study was to find the best wastewater collection system. In attempting to find this system the program had to select from an enormous number of possibilities the following:

1) A transportation route from a large number of feasible routes.
2) Depth-diameter combinations for each link in this route.
3) A set of treatment plant flows from numerous possible sets of flows.
4) The treatment plant efficiencies which will satisfy the dissolved oxygen standard.

It was found that the initial computer run, which placed no unnecessary restrictions on the circuited network would in general, find a good least cost solution in 5 to 7 iterations. Additional computer runs, at minimal extra computational cost could then be easily carried out by the user. These alternative runs should ensure that the lowest cost solution will be attained.

7.5 Recommendations

In networks where there are a large number of potential treatment facilities the user may not feel that placing several different restrictions on the initial circuited network will result in the lowest cost network being attained. Under these circumstances a method similar to that proposed by Whitlatch and ReVelle [37] could be employed.

Whitlatch and ReVelle used a heuristic location procedure based on the fact that the curve of cost vs. the number of treatment facilities in any one branching network, is unimodal. Heuristic techniques were then used in conjunction with optimal procedures to determine optimal location and number of treatment facilities. The initial estimate as to the number of plants involved in the optimal solution is determined by the user. This starting value is then compared to other alternatives. The best solution is continuously updated. The major drawback with this method is the large amount of computer time which is required in order to obtain the final solution. The large computational cost would however, be justifiable in light of the cost reduction which would be realized due to the lowest cost network being attained.

The accuracy of the results for the present study depend, in part on the size of the increment of water quality which is used. If the user wishes to accurately define the water quality standard then small increments will be required.

If there are a large number of potential treatment facilities and small increments of water quality are desired then a large amount of storage may be required. If storage becomes a problem then it would be necessary to employ a two pass system. In the first pass, coarse increments are used in order to define a region of water quality which should be searched in the second pass. The second pass searches this region using smaller increments to find the optimum treatment efficiencies. This two pass system will significantly reduce the amount of storage that will be required.

Leibman [18] showed that the type of problem considered in this study possesses the necessary properties to ensure that the optimum treatment efficiencies will be attained. To incorporate this procedure into the computer package parts of several subroutines would have to be rewritten.

The implementation of a variable water quality standard for each reach may prove to be a useful addition to the model. Typically, only certain parts of the river may require high levels of dissolved oxygen. A variable water quality standard could allow the user to assess the dissolved oxygen requirements for each reach. The water quality standard could then be relaxed for any reach which does not require a high dissolved oxygen concentration. This, in turn, may lead to a reduction in the treatment plant efficiencies which are required, and therefore an associated savings in the treatment plant costs would result.

It is this author's opinion that the study of the water quality aspects in this model be viewed only as a first attempt in solving this rather complex part of the overall problem.

Dissolved oxygen (DO) was selected as the parameter to represent water quality since adequate levels of DO are necessary to maintain satisfactory conditions for fish and other biological life in water. The effects of carbonaceous biological oxygen demand (BOD) of wastewater on the stream DO was considered in this study. The level of BOD is an important parameter in assessing the potential effects of pollutants on the concentration of DO in the stream. In addition, the costs for many treatment processes may be closely related to the percent BOD removal.

More recent[illegible] has been common to attempt to assess the effects of several pollutants on the water quality of the receiving body of water. The potential effect of total BOD (the sum of carbonaceous BOD and the nitrogeneous BOD) was discussed in this study. Other parameters which are commonly considered are phosphorus and several toxics.

Further research is therefore necessary to identify the pollutants which should be considered when assessing the water quality of the receiving body of water. It would then be necessary to determine the treatment processes which are commonly used to remove these pollutants.

A study of this nature would likely result in one of the following situations occurring:

a) The treatment processes which are required to remove the pollutants exhibit no (or very little) economies of scale (e.g. chemical precipitation to remove phosphorus).

b) The processes which remove specific pollutants are the same as those used to remove carbonaceous BOD and therefore no additional cost to the treatment of the wastewater will result.

c) A pollutant requires treatment processes which are different from those required to treat carbonaceous BOD. Furthermore, the processes required exhibit economies of scale.

If the results of the study showed that situations a) and/or b) held true for all pollutants then it would be reasonable to consider the treatment plant costs solely in terms of carbonaceous BOD.

Situation a) indicates that additional costs would result due to the treatment of a pollutant other than carbonaceous BOD. However, these additional costs could be treated as fixed costs. The total additional costs for treating the pollutant would be the same if it was treated at one central plant or several small plants. Situation b) is self explanatory in that the cost for treating the pollutant has already been accounted for.

If situation c) held true then it would be necessary to determine the feasibility of considering the treatment of more than one pollutant in the model.

BIBLIOGRAPHY

1. Adams, B.J., and Gemmell, R.S., "Water Quality Evaluation of Regionalized Wastewater Systems", Report No. 70, University of Illinois at Urbana-Champaign Water Resources Center, Illinois, August 1973.

2. Joint Committee of the American Society of Civil Engineers and the Water Pollution Control Federation,. "Design and Construction of Sanitary and Storm Sewers", ASCE Manual and Report 37, New York 1969.

3. Argaman, Y., Shamir, U., and Spivak, E., "Design of Optimal Sewerage Systems", ASCE, Journal of Environmental Engineering Division, Vol. 99, No. EES, October 1973, pp. 703-716.

4. Bartlett, R.E., "Pumping Stations for Water and Sewage", John Wiley and Sons, New York, 1974.

5. Bayer, M.B., "A Modelling Method for Evaluating Water Quality Policies in Nonserial River Systems", Water Resources Bulletin, Vol. 13, No. 6, December 1977, pp. 1141-1150.

6. Brill, E.D., and Nakamura M., "A Branch and Bound Method for use in Planning Regional Wastewater Treatment Systems", Water Resources Research, Vol. 14, No. 1, February 1978, pp. 109-118.

7. Chiang, C.H., and Lauria, D.T., "Heuristic Algorithm for Wastewater Planning", ASCE, Journal of the Environmental Engineering Division, Vol. 103, No. EE5, October 1977, pp. 863-875.

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

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

10. Deb, A.K., "Optimization in Design of Pumping Stations", Journal of the Water Pollution Control Federation, Vol. 50, February 1978, pp. 127-136.

11. Dobbins, W.E., "BOD and Oxygen Relationships in Streams", ASCE, Journal of Sanitary Engineering Division, Vol. 90, SA3, 53, June 1964.

12. "An Analysis of Construction Cost Experience for Wastewater Treatment Plants", Report No. 43019-76-002, U.S. Environmental Protection Agency, Washington, D.C., February 1976.

13. Graves, G.W., Hatfield, G.B., and Whinston, A.B., "Mathematical Programming for Regional Water Quality Management", Water Resources Research, Vol. 8, No. 2, April 1972, pp. 273-290.

14. Gupta, J.M., Agarwal, S.K. and Khanna, P., "Optimal Design of Wastewater Collection Systems", ASCE, Journal of the Environmental Engineering Division, Vol. 102, No. EE5, October 1976, pp. 1029-1041.

15. IBM System/360 and System/370 Fortran IV Language Reference Manual, Sample Problem 2.

16. Jarvis, J.J., Rardin, R.L., Unger, V.E., Moore, R.W., and Schimpeler, C.C., "Optimal Design of Regional Wastewater Systems: A Fixed-Charge Network Flow Model", Operations Research, Vol. 26, No. 4, August 1978, pp. 538-550.

17. Joeres, E.P., Dressler, J., Cho, C.C., and Falkner, C.H., "Planning Methodology for the Design of Regional Waste Water Treatment Systems", Water Resources Research, Vol. 10, No. 4, August 1974, pp. 643-649.

18. Leibman, J.C., "The Optimal Allocation of Stream Dissolved Oxygen Resources", Ph.D. Thesis, Cornell University Water Resources Center, 1965.

19. Liebman, J.C., and Lynn, W.R., "The Optimal Allocation of Stream Dissolved Oxygen", Water Resources Research, Vol. 2, No. 3, June 1966, pp. 581-591.

20. Lysne, D.K., "Hydraulic Design of Self-Cleansing Sewer Tunnels", ASCE, Journal of the Sanitary Engineering Division, 95, 17, 1967.

21. Main, W.M., "Design of Sanitary Sewer Systems by Dynamic Programming", M.Eng. Project, Department of Civil Engineering, McMaster University, Hamilton, Ontario, 1975.

22. O'Laoghaire, D.T., "A Coded Algorithm for Capacity Expansion of a Water Quality Management System", Water Resources Bulletin, Vol. 14, No. 4, August 1978, pp. 808-826.

23. ReVelle, C.S., Loucks, D.P., and Lynn, W.R., "Linear Programming Applied to water Quality Management", Water Resources Research, Vol. 4, No. 1, February 1968, pp. 1-9.

24. Rossman, L.A. and Liebman, J.C., "Optimal Regionalization of Wastewater Treatment for Water Quality Management, Research Report No. 89, Water Resources Center, University of Illinois at Urbana-Champaign, Ill., July 1974.

25. Rouse, H., "Fluid Mechanics for Hydraulics Engineers", Second Edition, Dover, New York, 1961.

26. Shields, A., "Anwending der Aenlichkeitsmechanik und der Turbulenzforsehung an die Geschiebebewegung" (Application of Similitude Mechanisms and Turbulence Research to Bed-Load Movement), Mitt. der Preuss. Versuchsanstalt fur Wasserbau and Schiffbau, No. 26, Berlin, 1936.

27. Shih, C.S., and DeFillippi, J.A., "Optimization Models for River Basin Water Quality and Waste Treatment Plant Design", Unpublished Report, Roy F. Weston Inc. Pennsylvania, 1969.

28. Siddall, J.N., "OPTISEP, Designer's Optimization Subroutines", ME/71/DSN/REPI, Faculty of Engineering, McMaster University, Hamilton, Ontario, 1974.

29. Smith, A.A., "C.E.P.L. = A Civil Engineering Program Library", McMaster University Bookstore, McMaster University, Hamilton, Ont. 1974.

30. Smith, A.A. and Tufgar, R.H., "Planning Regional Network Systems for Minimum Cost", Canadian Journal of Civil Engineering, Vol. 4, No. 2, June 1977, pp. 202-213.

31. Smith, R., "Cost of Conventional and Advanced Treatment of Wastewater", Journal of the Water Pollution Control Federation, Vol. 40, September 1968, pp. 1546-1574.

32. Streeter, H.W., and Phelps, E.B., "A Study of the Pollution and Natural Purification of the Ohio River", U.S. Public Health Service Bulletin 14-6, USPHS, Washington, D.C., 1925.

33. Tufgar, R.H., "Modelling of Regional Networks for Minimum Cost", M.Eng. Thesis, Department of Civil Engineering, McMaster University, Hamilton, Ontario, 1978.

34. 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.

35. "Design and Construction of Sanitary and Storm Sewers", WPCF Manual of Practice No. 9, Water Pollution Control Federation, Washington, D.C. (1960).

36. "Design and Construction of Sanitary and Storm Sewers", WPCF Manual of Practice No. 9, Water Pollution Control Federation, Washington, D.C. (1970).

37. Whitlach, E.E. Jr., and ReVelle, C.S., "Designing Regionalized Waste Water Treatment Systems", Water Resources Research, Vol. 12, No. 4, August 1976, pp. 581-591.

38. Wilcox, E.R., "A Comparitive Test of the Flow of Water in 8-in. Concrete and Vitrified Clay Sewer Pipe", University of Washington, Eng. Exp. Ser. Bull., 27, 1924.

39. Yao, K.M., "Functional Design of Sanitary Sewers", Journal of Water Pollution Control Federation, Vol. 48, July 1976, pp. 1772-1778.

40. Yarnell, D.L. and Woodward, S.M., "The Flow of Water in Drain Tile", U.S. Department Agriculture Bulliten, 845, 1920.

APPENDIX A

UNIMODAL COST VS. DIAMETER CURVE (GRAVITY FLOW)

Subroutine COSTT assumes that the cost vs. diameter curve is unimodal for a given flow.

The cost for a gravity link consists of pipe, excavation and manhole costs. The manhole cost accounts for only a small part of the link cost. Furthermore, a large percentage of the manhole cost is constant. Only when the upstream excavation depth increases significantly beyond the minimum cover does the manhole cost rise significantly. Therefore, in proving that the curve is unimodal the cost equation excluded manhole costs.

In order to show that the cost vs. diameter curve is indeed unimodal, two functions were calculated for all available diameters. The cost function is shown below. This cost includes all costs except shoring of excavation, restoring existing improvements and resorting existing permanent surfacing:

$$\text{COST} = \text{LENGTH} \times (9.73 + 0.097 \times (\text{CUT})^{1.6} + 0.826 \times (\text{DIA})^{2.387}) \qquad \text{(A.1)}$$

where,

LENGTH = length of pipe (ft)

CUT = average excavation (ft)

DIA = diameter of pipe (ft).

The following types of terrain were tested:

a) Relatively steep terrain, with a ground slope [(upstream ground elevation-downstream ground elevation)/length] equal to 0.02. The flow was set equal to 4.0 cfs.

b) Relatively shallow terrain, slope = 0.002. The flow was set equal to 2.0 cfs.

The upstream cover was set equal to minimum cover (8.0 ft) and the length of the link was set equal to 400 feet in both cases. The diameters considered ranged from 0.5 ft. to 2.5 ft. and increased in 0.167 foot increments.

Figures A.1(a) and A.1(b) both show distinct minimum. Clearly, each minima represents a combination of diameter and excavation (DOPT, EXCOPT) which results in the minimum cost for the link. The reduction in pipe costs which would occur if a diameter smaller than DOPT was chosen is more than offset by the increase in excavation costs which would occur. Conversely, choosing a diameter larger than DOPT results in a reduction in excavation costs which is less than the increase which results in the pipe costs.

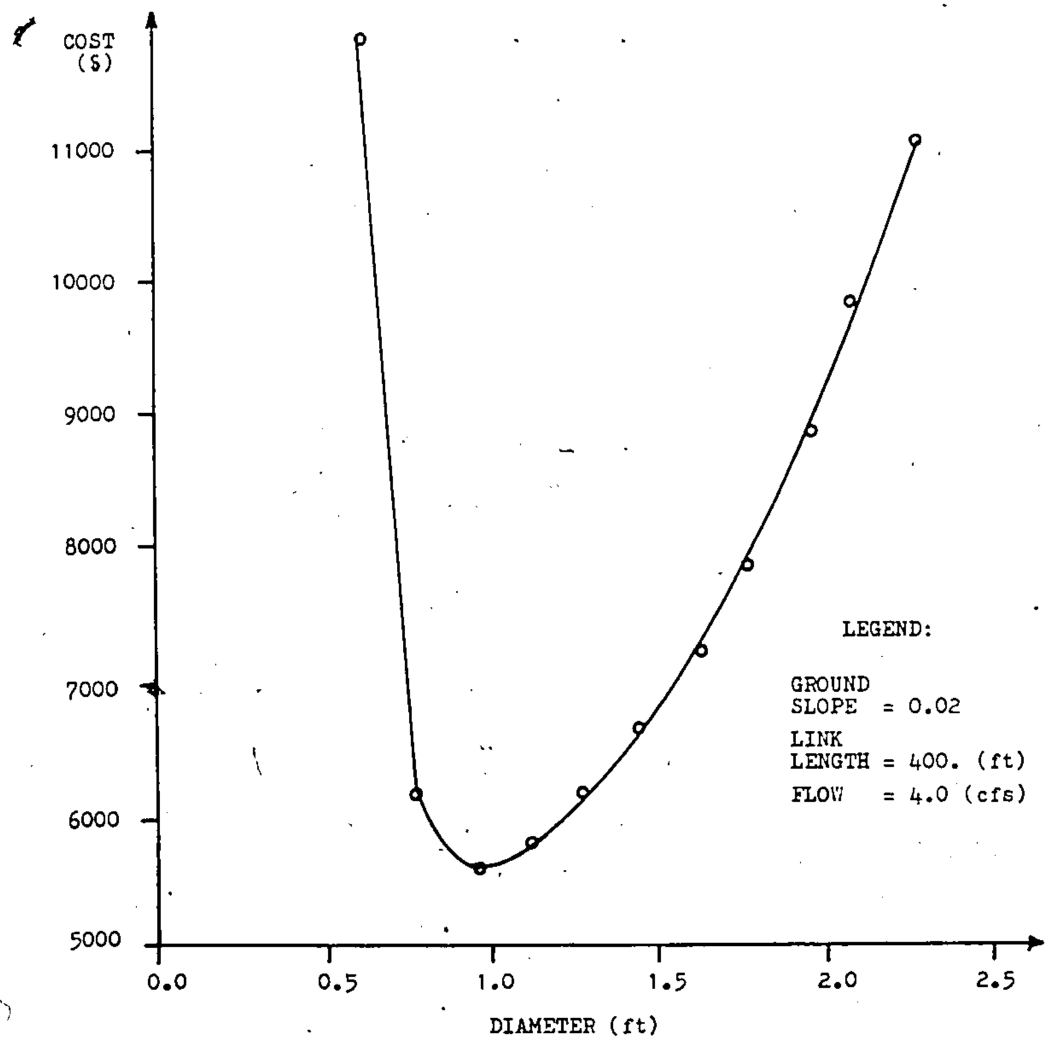


FIGURE A·1 (a) - UNIMODAL COST vs DIAMETER CURVE (GRAVITY FLOW, STEEP GROUND SLOPE)

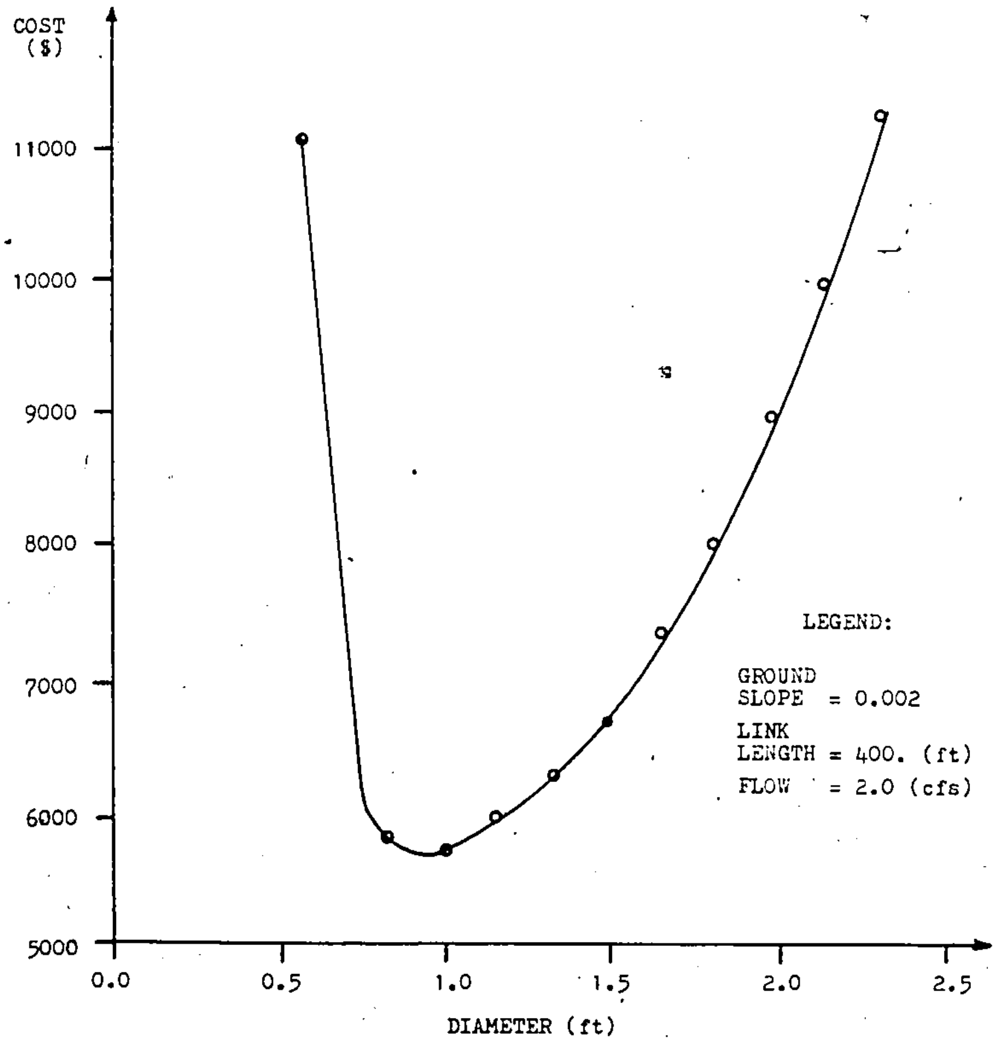


FIGURE A·1 (b) - UNIMODAL COST vs DIAMETER CURVE (GRAVITY FLOW, SHALLOW GROUND SLOPE)

APPENDIX B

PROPERTIES OF THE TRANSPORTATION COST VS. FLOW CURVE

For a curve to be strictly concave the value of the secant (COST/Q) for the cost vs. flowrate curve should continuously decrease as Q increases. Whether this will occur depends on the number of discrete diameters used and the nonlinearity of the cost curve.

Figure B.1 plots the link cost vs. diameter for an increasing flowrate. For a specific flowrate, Q_o, there will be one optimum diameter, d_o. Increasing Q_o slightly will, in general, increase the excavation costs with the same optimum diameter being used. Eventually the excess excavation costs will outweight the additional cost of installing a larger discrete diameter, d_{o+1}.

Excavation costs are proportional to the slope, which in turn is proportional to $(Q/kd^{8/3})^2$, for the full flow Mannings equation. If only discrete diameters are available then when the optimum diameter is increased the value of the slope and excavation costs will increase more slowly as the flowrate increases. If only a few discrete diameters are available, then the shape of the cost vs. flowrate curve may be of a "cuppy" nature as shown in Figure B.2.

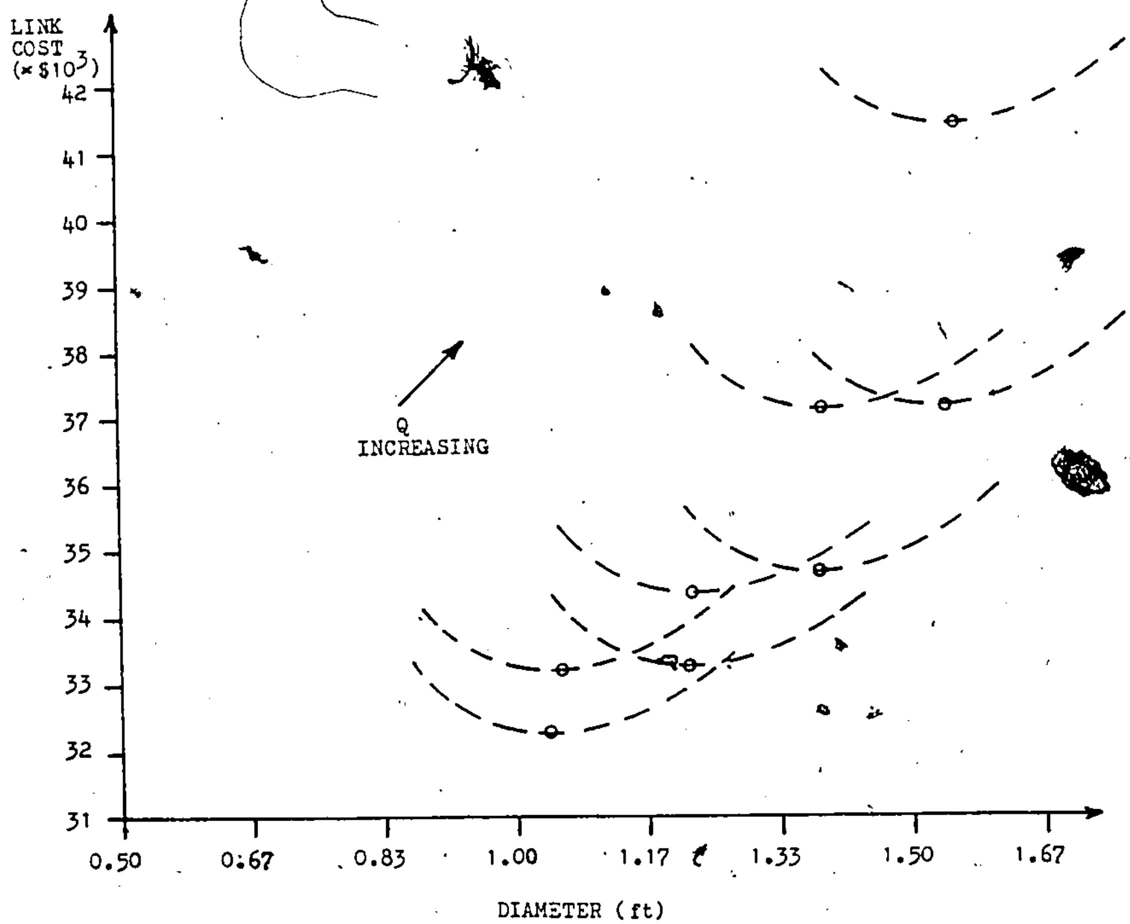


FIGURE B-1- LINK COST vs DIAMETER

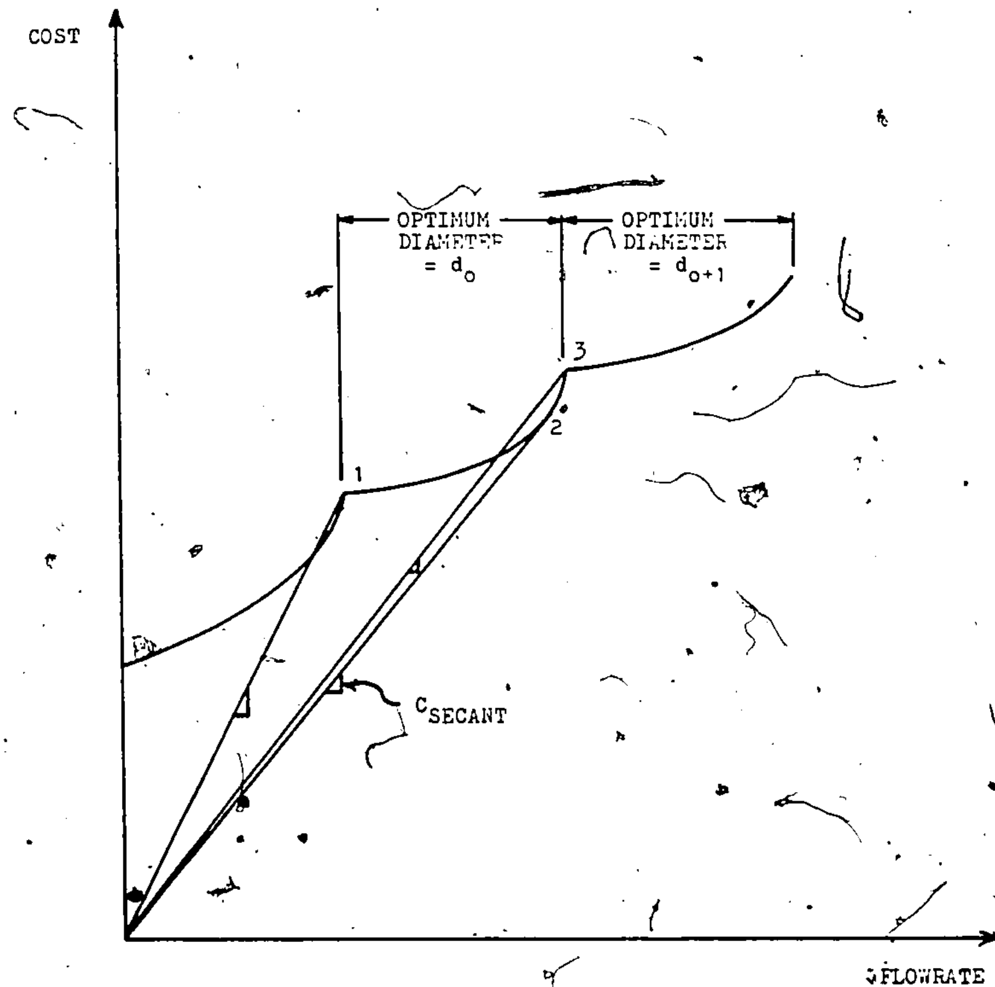


FIGURE B-2 - HYPOTHETICAL COST vs FLOWRATE CURVE (DISCRETE DIAMETERS)

Points 1, 2 and 3 illustrate the fact that as Q increases the value of the secant may not decrease. From points 1 to 2 the value of the secant does decrease, however, from points 2 to 3 the value of the secant increases, thus this curve is not strictly concave.

Figure B.3 illustrates a specific example. The increment between the discrete diameter was 2 inches. The cost function used was;

$$\text{COST} = \text{LENGTH} \times (9.73 + 0.097 \times \text{EXCAVATION}^{1.6} + 0.826 \times \text{DIA}^{2.387}) \qquad \text{(B.1)}$$

Although this curve may not seem to be strictly concave, secants taken at any two points show that the value of the secant does indeed decrease as the flowrate increases. This is due to the nonlinearity in the cost function.

Thus, the cost vs. flowrate curve will be strictly concave unless the curve is only slightly nonlinear and only a few discrete diameters are available. However, even if this situation does arise, the condition that the optimal solution would lie on a vertex would not be violated (see Appendix C). The extreme "cuppiness" of the cost vs. flowrate curve would be similar for all flow variables. The net result would be that the isocost lines would be

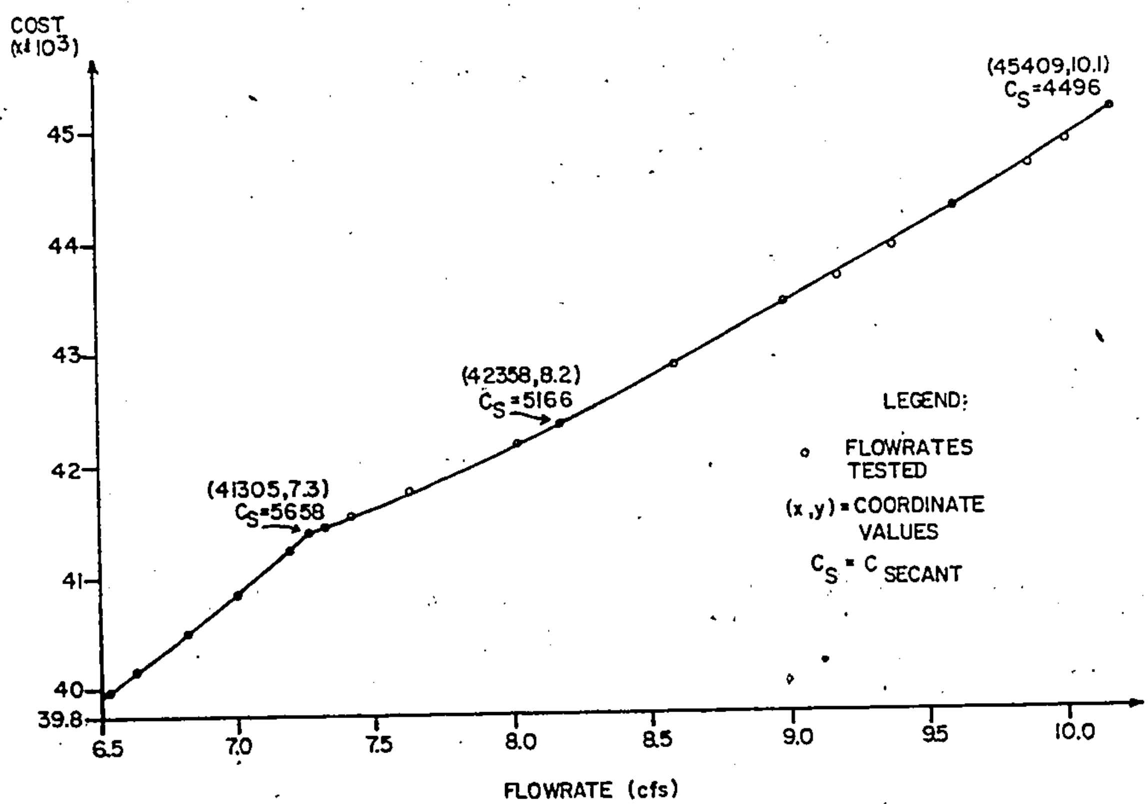


FIGURE B-3 - COST vs FLOWRATE CURVE FOR A SPECIFIC COST FUNCTION (DISCRETE DIAMETERS) unequally spaced as shown in Figure B.4. However, the isocost lines would still exhibit a concave shape on the Q_1, Q_2 plane and thus, the optimum solution would lie on a vertex.

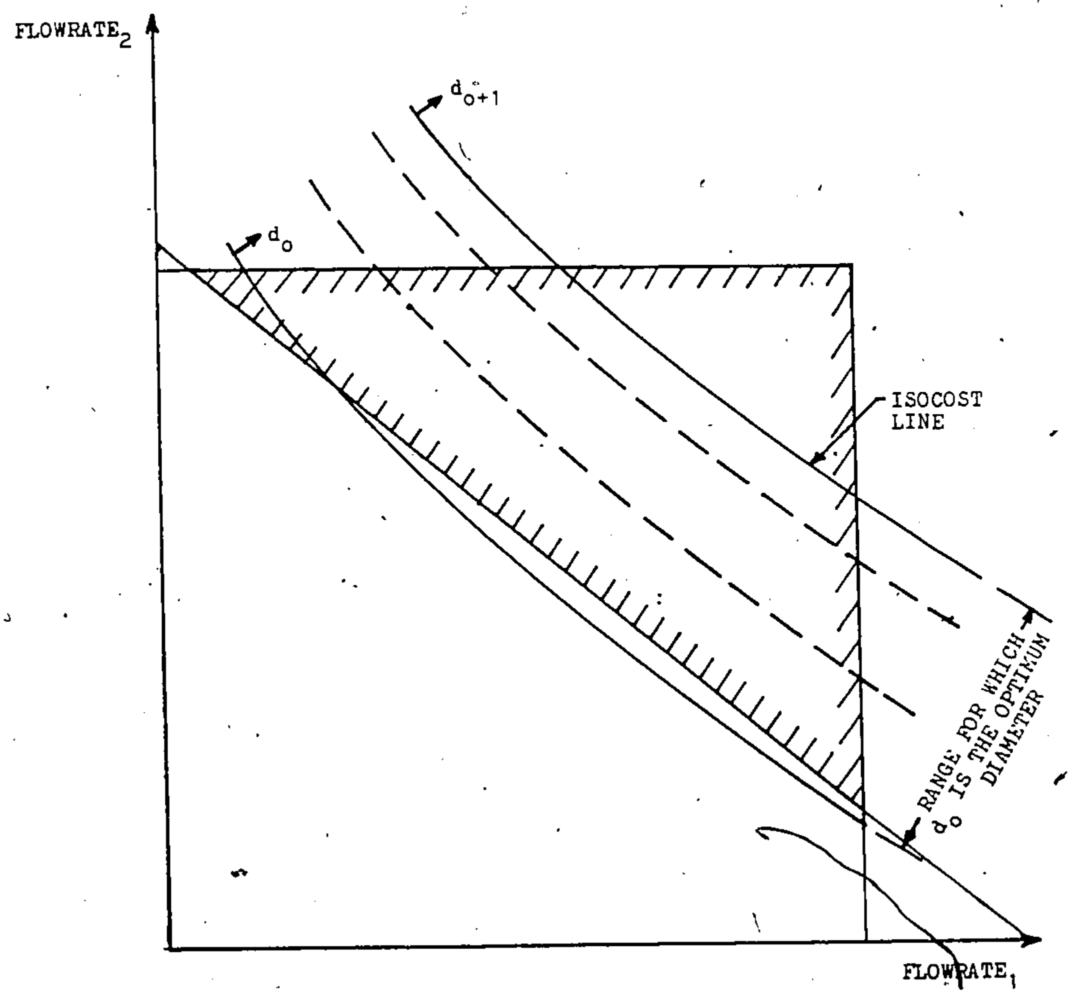


FIGURE B·4 - OBJECTIVE FUNCTION FOR A TWO VARIABLE PROBLEM (DISCRETE DIAMETERS)

APPENDIX C

PROPERTIES OF CONCAVE AND CONVEX PROGRAMMING

One property of linearly constrained problems is that the feasible space is convex, where convexity in a set of points means that the segment or line joining any two points in the set is also in the set.

For the case of minimization of a monotonic, convex objective function or maximization of a monotonic concave objective function, subject to these linear constraints, it can be shown that the solution must lie on the exterior of the feasible space but not necessarily on a vertex. However, any optimum found must be a global optimum. This type of problem is termed convex programming. To solve a convex progamming problem it is necessary to employ piecewise linearization of the objective function. This requires the introduction of extra variables and extra constraints.

For the case of maximization of a monotonic convex objective function or minimization of a monotonic concave objective function, subject to linear constraints, it can be shown that the solution must lie on the vertex of the feasible space, but that local optima may exist. This type of problem may be termed concave programming.

Figure C.1 shows a convex feasible region with a convex objective function. The case of minimization of a monotonic convex objective function is shown by point A. Point A represents a global optimum, but does not lie on a vertex. Point D, represents an optimum (not necessarily global) for the case of maximization of a convex objective function.

Figure C.2 shows a convex feasible region with a concave objective function. Point B is the optimum for the case of maximization of a monotonic concave function. This optimum is global, but does not, in general, lie on a vertex.

The case of minimization of a monotonic concave objective function is the case dealt with in this thesis. Figure C.2 clearly indicates the phenomena of local and global optima. At point P_1 a local optimum exists, since local gradients along edges P_1 to P_2 and P_1 to P_5 are both unattractive. Further exploration along P_1 to P_2 would reveal an eventual reduction of cost at P_2. P_2 represents the global optimum.

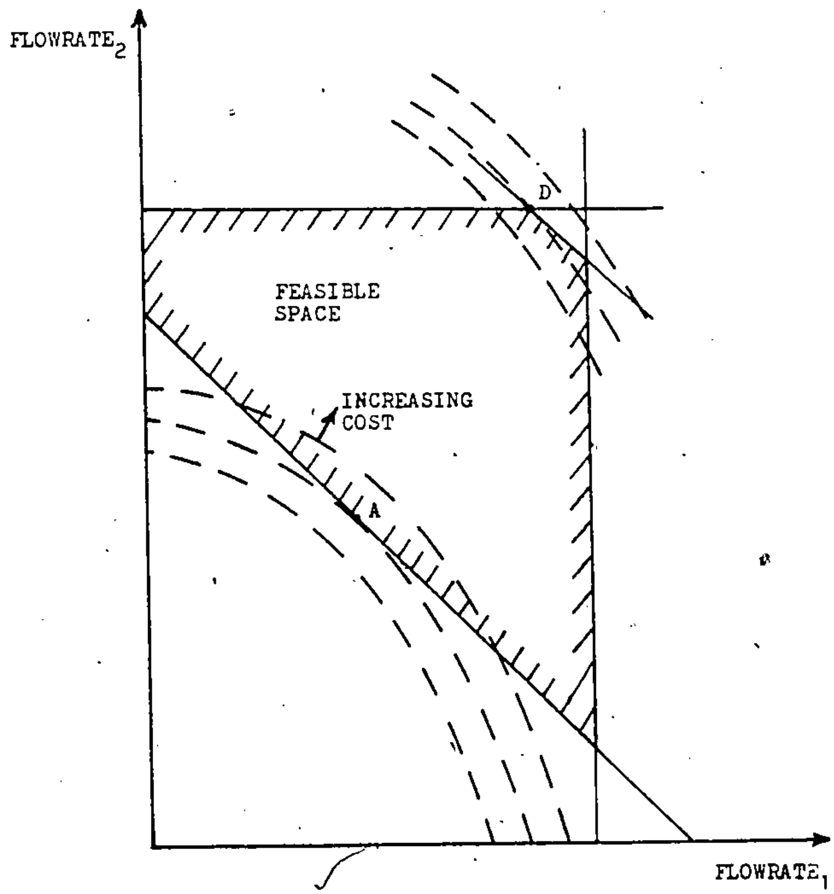


FIGURE C-1 — CONVEX OBJECTIVE FUNCTION

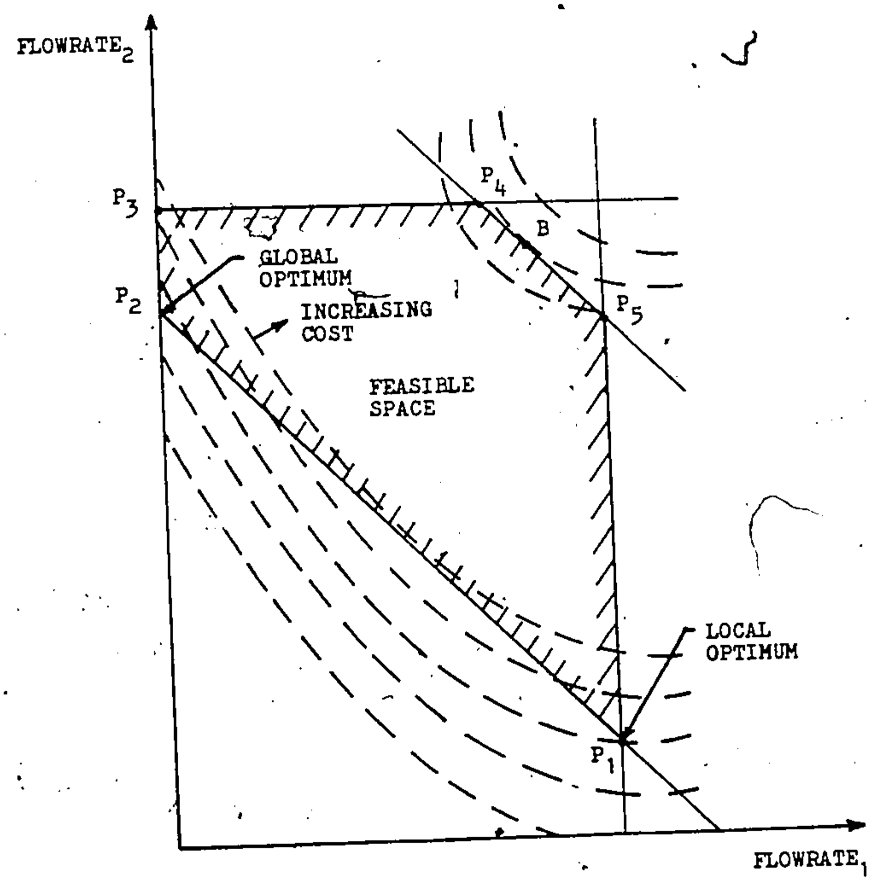


FIGURE C-2 - CONCAVE OBJECTIVE FUNCTION

APPENDIX D

UTILIZATION OF ASSIMILATIVE CAPACITY

A set of treatment plant flows is said to lead to better utilization of the assimilative capacity of the receiving body of water if, for a fixed total flow, a cost saving in the total plant cost occurs. There are two reasons why the overall total cost may be lower for certain sets of treatment plant flows.

The first reason is that a certain set of treatment plant flows may result in lower treatment efficiencies being required. This is illustrated in Figure D.1(a) and Figure D.1(b). These figures simplify the problem in that the treatment efficiency varies only at one plant. If the efficiencies varied at both plants (i.e. one plant efficiency decreased while the other increased) then the size of the treatment plants would have to be considered before it could be determined if a saving in cost had occurred.

The second reason is that combining two plants (and thereby benefiting from the economies of scale which generally occur) may result in a reduction in the total plant costs. Figure D.2(a) and Figure D.2(b) illustrate this point. It should be noted that in Figure D.2(a) and

Figure D.2(b) that the treatment efficiencies are the same. If the efficiency of the single plant (figure D.2(b)) had to be increased in order to meet the water quality standard then the reduction of scale may be outweighed by the additional cost due to the increased efficiency.

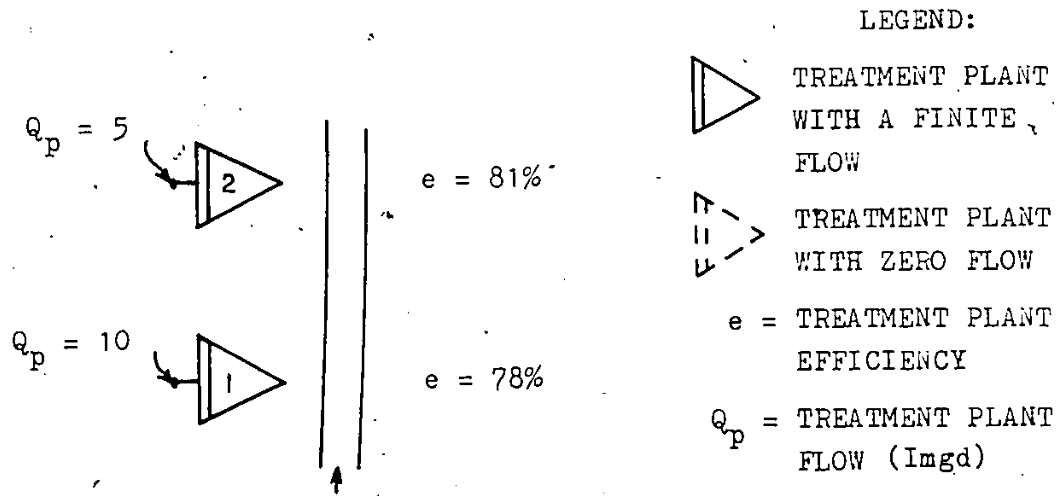


FIGURE D·1 (a) - PLANT CONFIGURATION REQUIRING HIGH TREATMENT PLANT EFFICIENCIES

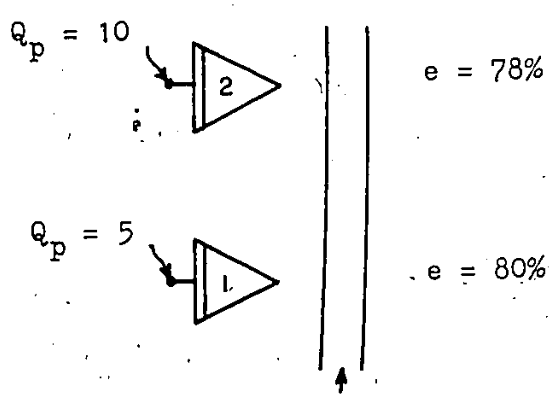


FIGURE D·1 (b) - PLANT CONFIGURATION REQUIRING LOW TREATMENT PLANT EFFICIENCIES

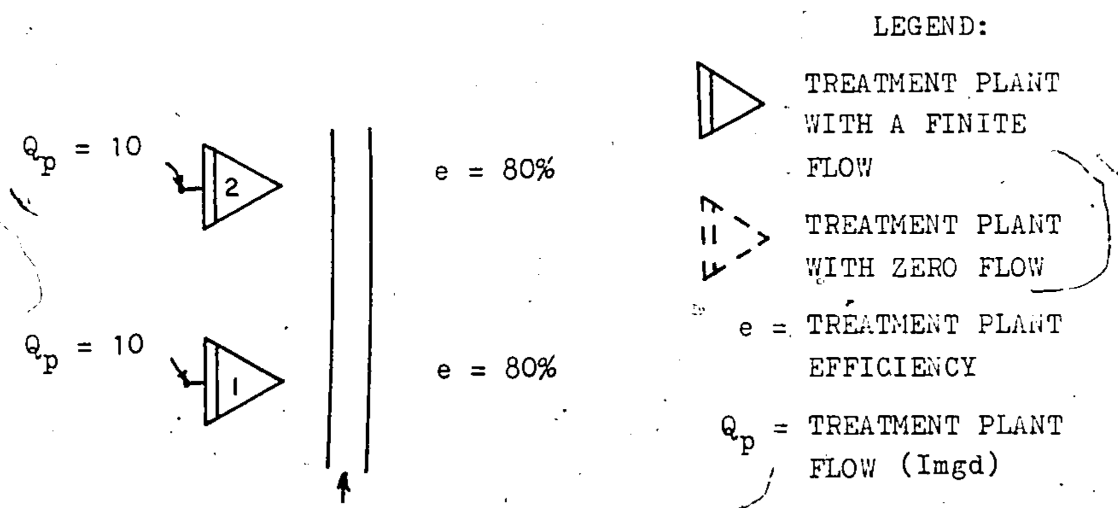


FIGURE D·2 (a) - DECENTRALIZED SOLUTION

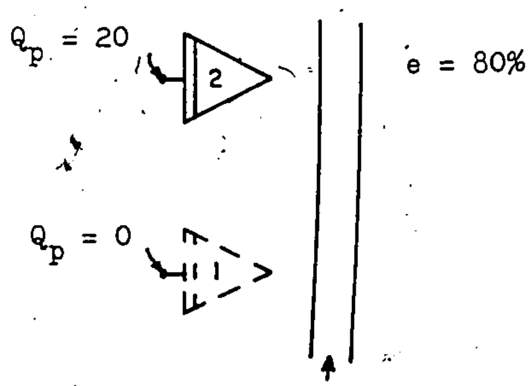


FIGURE D·2 (b) - SOLUTION EXHIBITING ECONOMIES OF SCALE

APPENDIX E

LISTING OF COMPUTER PROGRAM

```
      PROGRAM TST(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,
     +                 TAPE1,TAPE2)
C
C  DRIVER PROGRAM FOR THE NEW WASTEWATER NETWORK.
C
      DIMENSION QMAX(136),QMIN(136),STATE(40),XL(136),IFLAG(136)
      DIMENSION NAMES(4,40),CPEN(136),COST(16),A(40,136),C(136)
      DIMENSION DDISC(16),DOPT(136),NUS(136),NDS(136),CPROD(4)
      DIMENSION EUS(136),EDS(136),HP(136),BMAX(40),BMIN(40),NCENR(4)
      DIMENSION PINDUS(40),PINEND(40),ACRES(40),X(136)
      DIMENSION COMBEG(40),COMEND(40),POPBEG(40),POPEND(40)
      DIMENSION WA(40,40),WB(40),WC(40),WD(40),WE(40),WF(40)
      DIMENSION EFF(20),DECOPT(4,51),FOPT(4,51)
      DIMENSION R(4),XK(4),DEFICT(51),DSOL(4),RPLANT(4)
      DIMENSION TREACH(4),T(4),OPLANT(4),FUS(51,20)
      DIMENSION XINFNT(4),FD(51),FMAX(51),IEFF(51)
      DIMENSION FSOL(4),RTN(20),OPTF(4,51)
      LOGICAL ECHO
C
C  DATA FOR THE LINKS.
C
      XMNCOV=8.0
      NPIPES=16
      DMIN=0.667
      QAVG=0.000197
      NR1=1
      NR2=5
      NW=2
      REWIND NP1
      RINFLT=0.00186
      FLOW=5.0
      DATA DDISC/0.667,0.833,1.0,1.25,1.5,1.75,2.0,2.25,2.50,
     +      2.75,3.00,3.5,4.0,6.0,10.,20./
      READ(NP1,10)N,NCEN,L,TYPE
10    FORMAT(3I5,F10.1)
      NQ=2*L+NCEN
      L2=2*L
C
      CALL DEFINE(NAMES,STATE,BMAX,BMIN,NUS,NDS,XL,QMAX,NCENR,
     1              A,C,CPEN,N,NCEN,L,NQ,POPBEG,POPEND,COMBEG
     1,COMEND,RINDUS,RINEND,ACRES,QAVG,
     1RINFLT,NR1,NR2,TYPE,ECHO)
C
C  DATA FOR THE TREATMENT PLANTS. ALSO ASSIGN PENALTY TERMS HERE.
C
      QMAX(133)=-0.01
      QMAX(134)=-0.01
      QMAX(135)=-0.01
      QMAX(136)=-0.01
C  ADD RESTRICTIONS FOR THE TRTMT PLANTS IF DESIRED.
C  THERE IS NONE IN THIS EXAMPLE.
       NSTATE=51
       DALLOW=4.00
       ND=20
       SIGN=-1.0
       DS=0.080
       FO=2.0


      QRIV=300.0
      DATA EFF/0.35,0.45,0.55,0.60,0.65,0.70,0.75,0.80,0.84,0.87,
     +      0.89,0.90,0.92,0.94,0.95,0.96,0.97,0.98,0.99,0.999/
      DATA DEFICT/0.00,0.08,0.16,0.24,0.32,0.40,0.48,0.56,0.64,
     +      0.72,0.80,0.88,0.96,1.04,1.12,1.20,1.28,1.36,1.44,
     +      1.52,1.60,1.68,1.76,1.84,1.92,2.00,2.08,2.16,2.24,
     +      2.32,2.40,2.48,2.56,2.64,2.72,2.80,2.88,2.96,3.04,
     +      3.12,3.20,3.28,3.36,3.44,3.52,3.60,3.68,3.76,3.84,
     +      3.92,4.00/
      DATA XINFNT/35?.,350.,350.,350./
      DATA R/0.23,0.19,0.25,0.28/
      DATA XK/0.17,0.14,0.15,0.13/
      DATA TREACH/0.37,0.21,0.09,1.50/
      DATA T/8.0,7.0,7.5,5.5/
C
      CALL SOLVE(NQ,QMAX,QMIN,STATE,XL,XMNCOV,NPIPES,DMIN,DOPT,COST,FLOW
     +      ,TYPE,DDISC,EUS,EDS,HP,IFLAG,NUS,NDS,L2,NCENR,NCEN,NW,
     +      A,BMAX,BMIN,C,N,X,WA,WB,WC,WD,WE,WF,DEFICT,EFF,RPLANT,NSTATE,
     +      ND,SIGN,DECOPT,FOPT,FD,FMAX,DALLOW,QRIV,R,XK,TREACH,OPLANT,
     +      T,DS,XINFNT,FO,FUS,IEFF,RTN,OPTF,FSOL,DSOL,CPEN,CPROD)
C
C
      END
```

```
      SUBROUTINE DEFINE
     + (NAMES,STATE,STIP,STIPMN,NUS,NDS,XL,Q,NCENP,A,C,
     + CPEN,N,NCEN,L,NQ,POPBEG,POPEND,COMBEG,COMEND,RINDUS,
     + RINEND,ACRES,QAVG,RINFLT,NR1,NR2,TYPE,ECHO)
C
C ****************************************************************
C  ROUTINE READS NODE AND LINK DATA FROM TAPE NR1 AND
C  SETS UP ALL NECCESSARY WORKING ARRAYS        DIMENSION CPEN(NQ)
C    NAMES  = ARRAY OF SIZE (4,N) CONTAINING A NAME OF
C             UP TO 20 CHARS. FOR EACH OF N NODES.
C    STATE  = ARRAY OF SIZE (N) DEFINING STATE (E.G. ELEVATION)
C             OF EACH NODE FOR USE IN CALC. TRANSPORTN. COST.
C    STIP   = ARRAY OF SIZE (N) DEFINING YIELD (+VE) OR
C             ABSTRACTION (-VE) OF EACH NODE.
C    NUS    = ARRAY OF SIZE (NQ) DEFINING THE UPSTREAM
C             NODE NO. FOR EACH FLOW VARIABLE (ZERO FOR
C             THE LAST NCEN SLACK VARIABLES).
C    NDS    = ARRAY OF SIZE (NQ) DEFINING THE DOWNSTREAM
C             NODE NO. FOR EACH FLOW VARIABLE (ZERO FOR
C             THE LAST NCEN SLACK VARIABLES).
C    XL     = ARRAY OF SIZE (NQ) DEFINING THE LINK *LENGTH*
C             FOR EACH FLOW VARIABLE.  (MAY DIFFER IN D/S
C             AND U/S DIRECTIONS)
C    Q      = ARRAY OF SIZE (NQ) CONTAINING ON ENTRY THE
C             INITIAL APPROXIMATIONS AND ON EXIT THE OPTIMAL
C             VALUES OF THE FLOWRATES (INC. SLACK VARS.)
C    NCENP  = ARRAY OF SIZE (NCEN) TO HOLD COMPUTED
C             NODE NUMBERS OF PROCESSING NODES.
C    A      = ARRAY OF SIZE (N,NQ) TO HOLD THE STRUCTURAL
C             COEFFICIENTS COMPUTED BY ROUTINE  DEFINE.
C    CPEN   = ARRAY OF SIZE (NQ) TO HOLD COMPUTED VALUES
C             OF PENALTY COST COEFFICIENTS.
C    N      = INTEGER NO. OF NODES.
C    NCEN   = INTEGER NO. OF PROCESSING NODES IN SYSTEM.
C    L      = INTEGER NO. OF LINKS IN SYSTEM.
C    NQ     = 2*L+NCEN = NO. OF FLOW + SLACK VARIABLES.
C    NR1    = INPUT FILE DEVICE FOR NETWORK SYSTEM DATA.
C    NR2    = INPUT FILE DEVICE FOR COMMANDS.
C    TYPE   = NETWORK TYPE DESCRIPTOR SET TO
C             +1.0 FOR DISTRIBUTION NETWORKS
C             -1.0 FOR COLLECTION NETWORKS
C    ECHO   = LOGICAL VARIABLE SET .TRUE. IF ECHO
C             PRINTING OF COMMANDS ETC REQUD.
C ****************************************************************
      DIMENSION Q(NQ),XL(NQ)
       DIMENSION POPBEG(N),POPEND(N),COMBEG(N),COMEND(N),RINDUS(N)
      DIMENSION RINEND(N),ACRES(N),STIPMN(N)
      DIMENSION NUS(NQ),NDS(NQ),CPEN(NQ)
      DIMENSION STATE(N),STIP(N)
      DIMENSION NCENP(NCEN)
      DIMENSION NAMES(4,N)
      DIMENSION A(N,NQ)
      LOGICAL ECHO
      WRITE(6,5)
 5    FORMAT(35H IS ECHO PRINTING REQUD?  YES/NO    )
      READ(NR2,6) ANS
 6    FORMAT(A3)

      ECHO = .FALSE.
      IF(ANS.EQ.3HYES) ECHO = .TRUE.
C   CHECK INTEGERS
      IF(NQ.EQ.2*L+NCEN) GOTO 10
      WRITE(6,11)
 11   FORMAT(* INTEGERS MISMATCH IN DEFINE*)
 10   CONTINUE
C  READ DESCRIPTIONS OF NODES
      SUMO=0.0
      DO 100 JJ=1,N
      READ(NR1,105)I,POPBEG(I),POPEND(I),COMBEG(I),COMEND(I)
  105 FORMAT(I5,4F10.0)
      READ(NR1,106)I,RINDUS(I),RINEND(I),ACRES(I)
  106 FORMAT(I5,3F10.0)
      IF(ACRES(I).GT.0.0) GOTO 130
      STIPMN(I)=0.0
      STIP(I)=0.0
      GOTO 100
  130 CONTINUE
      PPBEGI=POPBEG(I)
      PPENDI=POPEND(I)
      CMBEGI=COMBEG(I)
      CMENDI=COMEND(I)
      RNDUSI=RINDUS(I)
      RNENDI=RINEND(I)
      ACRESI=ACRES(I)
      CALL CALCQ(PPBEGI,PPENDI,CMBEGI,CMENDI,RNDUSI,
     +           RNENDI,ACRESI,QAVG,RINFLT,STPMXI,STPMNI)
      STIP(I)=STPMXI
      STIPMN(I)=STPMNI
      SUMQ=SUMQ+STIP(I)
  100 CONTINUE
```

```
C
      DO 110 I=1,N
      IF(STIP(I).EQ.0.0) STIP(I)=+1.0*SUMQ
MULTIPLY BY 1 -- IGNORED
      IF(STIPMN(I).EQ.0.0) STIPMN(I)=-1.0*SUMQ
MULTIPLY BY 1 -- IGNORED
  110 CONTINUE
      DO 20 II=1,N
         READ(NR1,25) I,(NAMES(J,I),J=1,4),STATE(I)
      IF(ECHO)
     +   WRITE(6,25) I,(NAMES(J,I),J=1,4),STATE(I),STIP(I)
 25   FORMAT(I3,3X,4A5,F10.1,F10.3)
 20   CONTINUE
C  TEST FEASIBILITY OF SPECIFIED STIPULATIONS
      SIGMA=0.0
      DO 22 I=1,N
        SIGMA=SIGMA+STIP(I)
 22   CONTINUE
      IF(SIGMA*TYPE.GT.0.0) GOTO 24
      WRITE(6,23)
 23   FORMAT(24H STIPULATIONS INFEASIBLE)
      STOP
 24   CONTINUE
C  SET UP ARRAY NCENR(NCEN) DEFINING CENTRES
      IN=0

       DO 26 I=1,N
         IF(STIP(I)*TYPE.LE.0.0001) GOTO 26
         IN=IN+1
         IF(IN.LE.NCEN) GOTO 27
         WRITE(6,28)
 28      FORMAT(37H BAD DATA IN DEFINE  TOO MANY CENTRES)
         STOP
 27     CONTINUE
         NCENR(IN)=I
 26    CONTINUE
C  READ DESCRIPTIONS OF LINKS
       DO 30 II=1,L
         I2=II*2 - 1
         I2P1=I2+1
         READ(NR1,35)NUS(I2),NDS(I2),XL(I2),XL(I2P1),Q(I2),Q(I2P1)
       IF(ECHO)WRITE(6,35)NUS(I2),NDS(I2),XL(I2)
 35      FORMAT(2I5,3X,4F10.2)
         IF(XL(I2P1).LT.1.0E-10) XL(I2P1)=XL(I2)
       IF(Q(I2).LT.1.0E-10) Q(I2)=0.0
       IF(Q(I2P1).LT.1.0E-10) Q(I2P1)=0.0
         NUS(I2P1)=NDS(I2)
         NDS(I2P1)=NUS(I2)
 30    CONTINUE
C  SET NUS(),NDS()=ZERO FOR SLACK VARIABLES
       DO 38 I=1,NCEN
         II=I+2*L
         NUS(II)=NDS(II)=0
       XL(II)=0.0
 38    CONTINUE
C  SET STRUCTURAL COEFFS TO ZERO
       DO 40 I=1,N
         DO 40 J=1,NQ
           A(I,J)=0.0
 40    CONTINUE
C ASSIGN STRUCTURAL COEFFS FOR ALL FLOW VARIABLES
C  OUTFLOW=+VE:INFLOW=-VE
       L2=2*L
       DO 50 J=1,L2
         NUSI=NUS(J)
         NDSI=NDS(J)
         A(NUSI,J)=+1.0
         A(NDSI,J)=-1.0
 50    CONTINUE
C  ASSIGN STRUCTURAL COEFFS FOR SLACK VARIABLES
       DO 60 I=1,NCEN
         II=NCENR(I)
         J=L2 + I
         A(II,J) = TYPE
 60    CONTINUE
C  ASSIGN CONSISTENT INITIAL VALUES TO SLACK
C  VARIABLES BASED ON FIRST ESTIMATES OF Q()
C
       XNCEN=FLOAT(NCEN)
C
       DO 70 I=1,NCEN
         ISLACK=L2+I
         Q(ISLACK)=SUMQ*(XNCEN-1.00)/XNCEN

 70     CONTINUE
C  ZERO ALL COST PENALTY TERMS.
        DO 90 I=1,NQ
          CPEN(I) = 0.0
 90     CONTINUE
        RETURN
        END
```

```
C
C
      SUBROUTINE SOLVE(NQ,QMAX,QMIN,STATE,XL,XMNCOV,NPIPES,DMIN,DOPT,
     +  COST,FLOW,TYPE,DDISC,EUS,EDS,HP,IFLAG,NUS,NDS,L2,NCENR,NCEN,
     +  NW,A,BMAX,BMIN,C,N,X,WA,WB,WC,WD,WE,WF,DEFICT,EFF,
     +  BPLANT,NSTATE,ND,SIGN,DECOPT,FOPT,FD,FMAX,DALLOW,QPIV,R,XK,
     +  TREACH,QPLANT,T,DS,XINFNT,FO,FUS,IEFF,RTN,OPTF,FSOL,DSOL,CPEN,
     +  CPROD)
C
C
      DIMENSION QMAX(NQ),STATE(N),XL(NQ),IFLAG(NQ),DDISC(NPIPES)
      DIMENSION NUS(NQ),NDS(NQ),EUS(NQ),EDS(NQ),HP(NQ),DOPT(NQ),QMIN(NQ)
      DIMENSION NCENR(NCEN),A(N,NQ),BMAX(N),BMIN(N),C(NQ),X(NQ),CPEN(NQ)
      DIMENSION COST(NPIPES),WA(N,N),WB(N),WC(N),WD(N),WE(N),WF(N)
      DIMENSION EFF(ND),DECOPT(NCEN,NSTATE),FOPT(NCEN,NSTATE)
      DIMENSION R(NCEN),XK(NCEN),DEFICT(NSTATE),DSOL(NCEN)
      DIMENSION TREACH(NCEN),T(NCEN),QPLANT(NCEN),FUS(NSTATE,ND)
      DIMENSION XINFNT(NCEN),FD(NSTATE),FMAX(NSTATE),IEFF(NSTATE)
      DIMENSION FSOL(NCEN),RTN(ND),OPTF(NCEN,NSTATE),CPROD(NCEN)
      DIMENSION BPLANT(NCEN)
C   SINCE NSTAGE=NCEN , SAVE A PARAMETER IN THE ABOVE CALLING
C   STATEMENTS. NOW SET NSTAGE=NCEN.
      NSTAGE=NCEN
C
C   SET NPRINT=1 IF PRINTOUT REQUIRED FOR ALL DATA. SET NPRINT=0
C   IF ONLY FINAL RESULTS ARE REQUIRED.
C
      NPRINT=1
      NSTOP=5*N
      NNDEX=1
      IDATA=0
      ITER=0
C
C   ASSIGN A PENALTY COST TO ALL LINKS EXITING FROM A TRTMT CENTRE.
C   THUS, NO NEGATIVE TRTMT FLOWS SHOULD OCCUR.
      DO 133 I=1,L2
      IF(NUS(I).GT.NCEN) GO TO 133
      CPEN(I)=4.0E07
  133 CONTINUE
C
C   SET INITIAL VALUES FOR ALL FLOWS.
C
      DO 33 I=1,L2
      IF(QMAX(I).EQ.0.0) QMAX(I)=FLOW
   33 CONTINUE
      WRITE(6,34)QMAX
   34 FORMAT(3X,5F10.2)
C
 30   CONTINUE
C   ITERATION LOOP STARTS HERE.
      ITER=ITER+1
      IF(ITER.GT.10)   RETURN
C   COMPUTE COST COEFFICIENT.
C
      CALL SORT(NQ,N,QMAX,QMIN,NUS,NDS,IFLAG,EUS,EDS,HP,BMIN,STATE,XL,
     1                              COST,L2,DOPT,ITER,DDISC,NPIPES,DMIN,XM
     1NCOV,C)


      CALL CENCST(QMAX,NCENR,BMAX,C,N,NCEN,NQ,TYPE,30.0,0.09,0.05,
     +  DEFICT,EFF,NSTAGE,NSTATE,ND,SIGN,DECOPT,FOPT,FD,FMAX,
     +  DALLOW,QPIV,R,XK,TREACH,T,DS,XINFNT,FO,FUS,IEFF,CPROD,
     +  QPLANT,RTN,OPTF,FSOL,DSOL,NW,BPLANT)
C
C   SOLVE LINEAR PROGRAMMING PROBLEM.
C
C
C
C   ACCOUNT FOR ANY COST PENALTY TERMS BEFORE ENTERING SIMPLE.
      DO 642 I=1,NCEN
      ISLACK=L2+I
      C(ISLACK)=-C(ISLACK)
 642  CONTINUE
      DO 14 I=1,NQ
      C(I)=C(I)+CPEN(I)
  14  CONTINUE
      CALL SIMPLE(NQ,N,A,BMAX,C,NSTOP,IDATA,NNDEX,X,U,
     1                          WA,WB,WC,WD,WE,WF)
C
C  PRINTOUT INTERMEDIATE RESULTS.
      IF(NPRINT.EQ.0) GO TO 130
  35  CONTINUE
      WRITE(NW,10) ITER
  10  FORMAT(1H1,20X,*FOR ITERATION =*,I3,1X,*THE FOLLOWING*,/,
     +         20X,*LINKS COMPRISED THE BRANCHING NETWORK.*,//)
      WRITE(NW,20)
  20  FORMAT(22X,*FLOW*,4X,*DIA*,3X,*PUMP*,6X,*U/S*,6X,*D/S*,5X,
     +   *COST OF*,/,
     +  22X,*(CFS)*,3X,*(FT)*,4X,*HP*,5X,*INVERT*,3X,*INVERT*,
     +   5X,*LINK*,/,
     +  37X,*(FT)*,5X,*(FT)*,12X,*$1000.*,//)
```

```
      UOBJ=0.0
      DO 40 I=1,L2
      IF(QMAX(I).LT.0.01) GO TO 40
      UOBJ=UOBJ+QMAX(I)*C(I)
      CSTLNK=(QMAX(I)*C(I))/1000.
      WRITE(NW,41) NUS(I),NDS(I),QMAX(I),DOPT(I),HP(I),EUS(I),EDS(I),
     +  CSTLNK
41    FORMAT(11H FROM NODES,I3,3H TO,I3,F7.2,2X,F5.2,4F9.2)
40    CONTINUE
      WRITE(NW,42) UOBJ
42    FORMAT(//,10X,*TOTAL LINK COST=*,F15.2)
      WRITE(NW,43)
43    FORMAT(//,10X,*DATA FOR THE TRTMT PLANTS IS AS FOLLOWS*,//)
      WRITE(NW,44)
44    FORMAT(19X,*PLANT*,3X,*FLOW*,3X,*EFFICIENCY*,7X,*COST*,/,
     +         21X,*#*,4X,*(MGD)*,5X,*%*,12X,*$1000.*,//)
      TRTANS=0.0
      DO 50 J=1,NCEN
      ISLACK=L2+J
      QPROD=(TYPE*QMAX(J)-QMAX(ISLACK))*1.3464
      PERCNT=DSOL(J)*100.
      FSOLJ=FSOL(J)/1000.
      TRTANS=TRTANS+FSOLJ*1000.
      WRITE(NW,51) J,QPROD,PERCNT,FSOLJ
51    FORMAT(20X,I3,2X,F7.2,,4X,F6.2,2X,F12.2)
50    CONTINUE

      WRITE(NW,52)TRTANS
52    FORMAT(//,10X,*TOTAL PROCESSING COST=*,F15.2)
      TOTAL=UOBJ+TRTANS
      WRITE(NW,53)TOTAL
53    FORMAT(//,10X,*TOTAL SYSTEM COST=*,F15.2,/////)
130   CONTINUE
C
C  CHECK TO SEE IF ERRORS ARE ACCEPTABLE.
C
      QLIMIT=0.001
      IERROR=0
      DO 200 I=1,NQ
      ERR=ABS(X(I)-QMAX(I))
      IF(ERR.GT.QLIMIT) IERROR=1
200   CONTINUE
C
C  UPDATE ALL QMAX(I) VALUES.
C
      DO 350 I=1,NQ
      QMAX(I)=X(I)
350   CONTINUE
C
      IF(IERROR.GT.0) GO TO 30
      IF(NPRINT.EQ.0) NPRINT=2
      IF(NPRINT.GT.1) GO TO 35
      RETURN
      END
```

```
      SUBROUTINE SORT(NQ,N,Q,DMIN,NUS,NDS,IFLAG,EUS,EDS,HP,BMIN,
     1                             STATE,XL,COST,L2,DOPT,ITER,DDISC,NPIPES,DMI
     1N,XMNCOV,C)
C
C   ITER IS A COUNTER TO FIND OUT IF THIS IS THE FIRST
C   EVAL OF THE COST COEFF C(I). IF SO THEN ITER=1
C   AND ALL Q(I)=0.1 (EXCEPT THE SLACK VAR). IF NOT,THEN
C   SUBROUTINE SIMPLEX WILL HAVE SENT BACK (N-1) FINITE
C   VALUES OF Q WHICH WILL BE USED TO FORM A BRANCHED
C   NETWORK.THE OTHER LINKS WILL FORM THE REDUNDANT NETWORK.
C
      DIMENSION DOPT(NQ),Q(NQ),NUS(NQ),NDS(NQ),IFLAG(NQ),COST(NPIPES)
      DIMENSION EUS(NQ),EDS(NQ),STATE(N),XL(NQ),C(NQ),DDISC(NPIPES)
      DIMENSION QMIN(NQ),BMIN(N),HP(NQ)
C
C   CHECK TO SEE IF ITER=1
C
      IF(ITER.GT.1) GO TO 5
      DROP=0.0
      USDIA=DMIN
C
      DO 6 I=1,L2
      NUSI=NUS(I)
      NDSI=NDS(I)
      GUSI=STATE(NUSI)
      GDSI=STATE(NDSI)
      EUSI=STATE(NUSI)-XMNCOV
      EDSI=STATE(NDSI)-XMNCOV
      XLI=XL(I)
      QI=Q(I)
      QMINI=1.00*CI
MULTIPLY BY 1 -- IGNORED
      CALL CALCCI(EUSI,USDIA,QI,QMINI,XLI,I,DDISC,NPIPES,COST,
     1                                  GUSI,GDSI,XMNCOV,HPI,EDSI,XMNCST,DOP
     1TI)
      HP(I)=HPI
      EUS(I)=EUSI
      EDS(I)=EDSI
      DOPT(I)=DOPTI
      C(I)=XMNCST/Q(I)
6     CONTINUE
      RETURN
5     CONTINUE
C
C   FOR ANY BRANCHED NETWORK THE SEQUENCE BY WHICH EACH C(I)
C   VALUE SHOULD BE CALCULATED WILL BE DETERMINED. THEN THE C(I)
C   VALUES FOR THE REDUNDANT LINKS WILL BE DETERMINED.
C
      DO 3 I=1,L2
3     DOPT(I)=DMIN
C
      I=ISUM=IFINQ=0
C
C   CALC.THE NUMBER OF FINITE FLOWS RETURNED FROM SIMPLEX.
C
      DO 10 M=1,L2
      IFLAG(M)=0


      IF(Q(M).EQ.0.0) GO TO 10
      IFINQ=IFINQ+1
10    CONTINUE
20    CONTINUE
      I=I+1
      IF(I.GT.L2)I=1
      IF(Q(I).EQ.0.0) GO TO 20
C
C     IF IFLAG=1 THEN THIS LINK HAS ALREADY
C   BEEN ACCOUTNED FOR.
C
      IF(IFLAG(I).EQ.1) GO TO 20
C
C   DETERMINE WHETHER OR NOT LINK(I) HAS ANY U/S LINKS AND
C   WHETHER OR NOT THESE U/S LINKS HAVE A FINITE Q VALUE.
C   ALSO, FIND THE SMALLEST DIA. AND THE LOWEST ELEVATION
C   (OF ANY BRANCHED LINK)  ENTERING LINK(I).
C
      QTOT=0.0
      DO 50 M=1,L2
      IF(NDS(M).NE.NUS(I)) GO TO 50
      IF(Q(M).EQ.0.0) GO TO 50
      QTOT=QTOT+Q(M)
50    CONTINUE
```

```
C
      IF (QTOT.GT.0.0) GO TO 60
C
C     THIS IS AN U/S LINK.THUS EUSI=STATE(NUSI)-XMNCOV
C     AND USDIA=DMIN (FOR STARTERS ANYWAY).
C
      USDIA=DMIN
      IFLAG(I)=1
      ISUM=ISUM+1
      NUSI=NUS(I)
      NDSI=NDS(I)
      GUSI=STATE(NUSI)
      GDSI=STATE(NDSI)
      EUSI=STATE(NUSI)-XMNCOV
      EDSI=STATE(NDSI)-XMNCOV
      XLI=XL(I)
      QI=Q(I)
      QMIN(I)=BMIN(NUSI)
      QMINI=QMIN(I)
C
      CALL CALCCI(EUSI,USDIA,QI,QMINI,XLI,I,DDIST,NPIPES,COST,
     1                  GUSI,GDSI,XMNCOV,HPI,EDSI,XMNCST,DOPTI)
C
C     DOPTI WILL BE SENT BACK UP FROM SUB. COSTT.
C     THE DIA. SENT BACK UP MAY BE USED FOR THE U/S DIA.
C     CONSTRAINT FOR THE OTHER LINKS. EDSI WILL BE SENT BACK UP
C     FROM CALCCI. IT ALSO WILL BE USED AS A CONSTRAINT FOR
C     THE D/S LINKS.
C
      HP(I)=HPI
      DOPT(I)=DOPTI
      EUS(I)=EUSI
      EDS(I)=EDSI

      C(I)=XMNCST/QI
      IF(ISUM.EQ.IFINQ) GO TO 999
      GO TO 20
C
60    CONTINUE
C
C     THIS IS NOT AN U/S LINK AND IT MUST BE DETERMINED
C     WHETHER OR NOT ALL U/S BRANCHED LINKS (Q(I).GT.0.0)
C     HAVE BEEN CALCULATED.ALSO, IF THE U/S LINKS HAVE BEEN
C     CALCULATED THE DIA. AND ELEV. CONSTRAINTS DUE TO
C     THESE U/S LINKS MUST BE CONSIDERED.
C
      DO 70 M=1,L2
      IF(NDS(M).NE.NUS(I)) GO TO 70
      IF(Q(M).EQ.0.0) GO TO 70
      IF(IFLAG(M).NE.1) GO TO 20
70    CONTINUE
C     ALL U/S LINKS HAVE BEEN ACCOUNTED FOR. NOW TAKE INTO
C     CONSIDERATION THE DIA. AND ELEV. CONSTRAINTS DUE TO THE
C     U/S LINKS WITH FINITE Q VALUES.NOTE,TO CALC. USDIA AND
C     EDS(NDSM) ALL U/S LINKS MUST HAVE BEEN CALCULATED.
C
      USDIA=DMIN
      QMINI=0.0
      EDS2=50000.
      DO 90 M=1,L2
      IF(NDS(M).NE.NUS(I)) GO TO 90
      IF(Q(M).EQ.0.0) GO TO 90
C
C     Q(M)>0.0. FURTHERMORE,TO GET THIS FAR QMIN(M) IS KNOWN.
C
      QMINI=QMINI+QMIN(M)
      IF(EDS2.LT.50000.) GO TO 82
C
C     THIS IS THE FIRST (AND POSSIBLY THE ONLY) U/S LINK
C     WITH A FINITE Q VALUE.
C
      EDS2=EDS(M)
      USDIA=DOPT(M)
      GO TO 90
C
82    CONTINUE
C
C     MORE THAN ONE U/S LINK WITH A FINITE FLOW HAS BEEN ENCOUNTERED.
C
      IF(EDS(M).GT.EDS2) GO TO 86
      EDS2=EDS(M)
86    IF(DOPT(M).LT.USDIA) GO TO 90
      USDIA=DOPT(M)
90    CONTINUE
      EUS(I)=EDS2
C
      IFLAG(I)=1
      ISUM=ISUM+1
      EUSI=EUS(I)
      NUSI=NUS(I)
      NDSI=NDS(I)
```

```
      GUSI=STATE(NUSI)
      GDSI=STATE(NDSI)
      EDSI=STATE(NDSI)-XMNCOV
      XLI=XL(I)
      QI=Q(I)
      QMINI=QMINI+BMIN(NUSI)
C
      CALL CALCCI(EUSI,USDIA,QI,QMINI,XLI,I,DDIST,NPIPES,COST,
     1                          GUSI,GDSI,XMNCOV,HPI,EDSI,XMNCST,DOPTI)
C
      HP(I)=HPI
      EUS(I)=EUSI
      EDS(I)=EDSI
      DOPT(I)=DOPTI
      C(I)=XMNCST/QI
C
C  FOR USE FURTHER D/S.
C
      QMIN(I)=QMINI
      IF(ISUM.EQ.IFINQ) GO TO 999
      GO TO 20
999   CONTINUE
C
C  THIS PART OF THE SUBROUTINE DEALS WITH THE REDUNDANT
C  LINKS. THIS PART, WILL FOR ANY REDUNDANT LINK FIND THE
C  ELEV. AND DIA. CONSTRAINTS AT THE U/S NODE, AS WELL AS
C  THE TOTAL FLOW ENTERING THE U/S NODE FROM THE BRANCHED LINKS.
C
      DO 201 I=1,L2
      IF(Q(I).GT.0.0) GO TO 201
      QMINI=QTOTT=0.0
      NUSI=NUS(I)
      EDS2=STATE(NUSI)-XMNCOV
      USDIA=DMIN
C
      DO 501 M=1,L2
      IF(NUS(M).EQ.NUS(I))QTOTT=QTOTT+Q(M)
      IF(Q(M).EQ.0.0) GO TO 501
      IF(NDS(M).NE.NUS(I)) GO TO 501
      QMINI=QMINI+QMIN(M)
      IF(NDS(M).EQ.NUS(I).AND.NUS(M).EQ.NDS(I)) GO TO 501
C
C  WON*T GET THIS FAR UNLESS THERE IS AN U/S LINK
C  WITH A FINITE Q VALUE
C
      IF(EDS2.LT.50000.0) GO TO 102
      EDS2=EDS(M)
      USDIA=DOPT(M)
      GO TO 501
102   CONTINUE
      IF(EDS(M).GT.EDS2)GO TO 96
      EDS2=EDS(M)
96    IF(DOPT(M).LT.USDIA) GO TO 501
      USDIA=DOPT(M)
501   CONTINUE
      EUS(I)=EDS2

      NUSI=NUS(I)
      NDSI=NDS(I)
      GUSI=STATE(NUSI)
      GDSI=STATE(NDSI)
      EUSI=EUS(I)
      EDSI=STATE(NDSI)-XMNCOV
      XLI=XL(I)
      QI=QTOTT
      QMINI=QMINI+BMIN(NUSI)
      IF(QI.GT.0.0) GO TO 504
      QI=QMINI=0.1
504   CONTINUE
C
      CALL CALCCI(EUSI,USDIA,QI,QMINI,XLI,I,DDIST,NPIPES,COST,
     1                          GUSI,GDSI,XMNCOV,HPI,EDSI,XMNCST,DOPTI)
C
      HP(I)=HPI
      EUS(I)=EUSI
      DOPT(I)=DOPTI
      EDS(I)=EDSI
      C(I)=XMNCST/QI
C
C  RETURN AVALUE OF Q(I)=0.00 SO AS THE OTHER LINKS DO NOT
C  ASSUME THAT THIS WAS A BRANCHED LINK.
C
      Q(I)=0.00
201   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CALCCI(EUSI,USDIA,QMAX,QMIN,XLI,I,DDISC,NPIPES,COST,
     1                          GUSI,GDSI,XMNCOV,HPI,EDSI,XMNCST,DOP
     1TI)
      DIMENSION DDISC(NPIPES),COST(NPIPES)
C
C  ORGANIZATIONAL ROUTINE WHICH WILL CALL SUBROUTINES VMINN.
C  VMAXX AND COSTT. CALCCI SENDS BACK,TO SUB. SOPT ,XMNCST.
C  EDSI AND DOPTI.
C
C
C  CALC. SAVAILWHERE SAVAIL=(EUSI-EDSI)/XLI. ANY SLOPE LESS
C  THAN THIS WILL VIOLATE THE MIN. COVER CONSTRAINT.(UNLESS A
C  A DROP MANHOLE IS USED)
C
      SAVAIL=(EUSI-EDSI)/XLI
C
C  FINC IL CORRESPONDING TO THE DISCRETE DIA. WHICH IS EQUAL
C  TO USDIA.USE THIS AS THE STARTING DIA. (SINCE ANY DIA.
C  DIA. SMALLER THAN THIS IS NOT ALLOWED DUE TO THE DIA.
C  PROGRESSION CONSTRAINT) .TEST TO SEE IF THE CONSTRAINTS
C  ARE SATISFIED,THEN INCREASE DIA.(SIZPIP) UNTIL AN INCREASE
C  IN COST OCCURS.
      DO 110 J=1,NPIPES
      IF(DDISC(J).EQ.USDIA) GO TO 100
110   CONTINUE
100   IL=J-1
101   CONTINUE
      IL=IL+1
      SIZPIP=DDISC(IL)
C
      CALL VMINN(QMIN,SAVAIL,SIZPIP,0.04,114.31,63.65,S,SMNLMT)
C
C
      CALL VMAXX(QMAX,10.0,SIZPIP,S,SMNLMT,IOK,S2)
C
      IKOUNT=0
      IF(DDISC(IL).EQ.USDIA) IKOUNT=1
C
      CALL COSTT(EUSI,GUSI,S,S2,IKOUNT,IOK,NPIPES,IL,XLI,I,
     1                    GDSI,DDISC,DROP,COST,SIZPIP,XMNCST,EDSI,DOP
     1TI)
C
C  COMPARE MIN. COST OF LINK WITH GRAVITY FLOW TO MIN. COST OF
C  LINK WITH PUMP-FORCE MAIN COMBINATION. COMPARE ONLY IF THE
C  D/S COVER FOR THE OPTIMUM DIA. IN THE GRAVITY FLOW CASE
C  IS > (20.+XMNCOV). OTHERWISE USE GRAVITY FLOW.
C
      IF(IOK.NE.2) GO TO 101
      HPI=0.0
      IF((GDSI-EDSI).LT.(0.+XMNCOV)) RETURN
CONSTANT TERM OF ZERO -- IGNORED
      CALL STACST(EUSI,GUSI,GDSI,QMAX,QMIN,XLI,NPIPES,DDISC,15.0,
     1                    0.04,114.31,63.65,XMNCOV,0.55,30.,0.09,0.05
     1,COST,                                HPI,XCOST,EDSPMP,PMPDIA)
      IF(XCOST.LT.XMNCST) GO TO 104
      HPI=0.0
      RETURN

104   XMNCST=XCOST
      EUSI=GUSI-XMNCOV
      EDSI=EDSPMP
      DOPTI=PMPDIA
      RETURN
      END
```

```
      SUBROUTINE VMINN(QS,SAVAIL,SIZPIP,T,CBYN,W,S,SMNLMT)
C
C   VMIN=MIN. FULL FLOW VEL. IN THE PIPE, BASED ON THE
C   CRITICAL SHEAR STRESS APPROACH. THE MIN. VEL. WILL INCREASE
C   WITH DIAMETER.
C   T= SPECIFIED MIN. BOUNDARY SHEAR STRESS.(CRITICAL SHEAR STRESS)
C   TYPICAL VALUES ARE 0.03 TO0.04 LB/SQ. FT.
C   W= SPECIFIC WEIGHT OF WASTEWATER (1.02*62.4 LB/CU. FT.)
C   QS=MINIMUM DESIGN FLOW AT BEGINNING OF DESIGN PERIOD.
C   QF=VMIN*AREA
C   SC=SLOPE OF SEWER CARRYING QS AND AS SELF CLEANSING AS
C   SEWER FLOWING FULL WITH VELOCITY VMIN.
C   SF=REQ*D SLOPE TO ATTAIN VMIN(FULL FLOW COND.)
C   QS IS KNOWN. IT IS REQ*D TO FIRSTLY, FIND SF WHICH WILL
C   RESULT IN V=VMINN, THEN FIND SC.
C   SC WILL THEN BE COMPARED TO SAVAIL.
C
      VMIN=CBYN*(SIZPIP/4.0)**0.1667*(T/W)**0.5
      QF=VMIN*3.1416*SIZPIP*SIZPIP/4.0
      SF=(QF/(35.71*SIZPIP**2.667))**2.0
      IF((QS/QF).GT.0.5) GO TO 300
C
C  CALCULATE SC. SC WILL DEFINE THE MINIMUM SLOPE AT WHICH THIS
C  PIPE CAN BE LAID. ANY SLOPE LESS THAN SC WOULD VIOLATE THE
C  MIN. VEL. CONSTRAINT WHICH IS BASED ON THE BEGINNING OF THE
C  DESIGN PERIOD.
      IF((QS/QF).GE.0.19) GO TO 310
      SC=SF*(9.79-269.29*(QS/QF)+3540.58*(QS/QF)**2.0-
     1               2)348.2*(QS/QF)**3.0+41872.5*(QS/QF)**4.0)
      GO TO 320
C
310   SC=SF*(2.23-6.26*(QS/QF)+12.46*(QS/QF)**2.0-
     1               13.19*(QS/QF)**3.0+5.74*(QS/QF)**4.0)
      GO TO 320
300   CONTINUE
C
      IF((QS/QF).LT.1.0) GO TO 333
C
C  PIPE WOULD NOT BE ABLE TO CARRY FLOW QS AT SLOPE SF. THUS
C  INCREASE SF SO AS CAPACITY REQUIREMENTS ARE MET.PIPE WILL FLOW
C  JUST FULL. (SET QF=QS).
C
      QF=QS
      SF=(QF/(35.71*SIZPIP**2.667))**2.0
      SC=SF
      GO TO 328
C
C  SC WOULD BE LESS THAN SF AND THUS SF WILL BE USED.
C
333   CONTINUE
C
C  SC WOULD BE LESS THAN SF.THUS SF WILL BE USED.
      SC=SF
320   CONTINUE
C
C  COMPARE SC TO SAVAIL. SC IS DEFINED AS THE REQ*D SLOPE
C  TO ATTAIN A VEL. AS SELF CLEANSING AS A SEWER FLOWING FULL
C  AT VMIN.IF SAVAIL > SC THEN THE VEL. IN THE PIPE WILL BE
C  GREATER THAN THE REQUIRED SELF CLEANSING VELOCITY.
C  IF SAVAIL.LT. SC THEN THE VEL. IN THE PIPE WILL BE
C  INSUFFICIENT AND THUS SAVAIL MUST BE SET EQUAL TO SC.
C
328   IF(SAVAIL.GT.SC) GO TO 333
C
C  SAVAIL IS .LE. SC
C
      S=SC
      SMNLMT=SC
      RETURN
333   CONTINUE
C
C  SAVAIL.GT.SC. SET S=SAVAIL SINCE THE MINIMUM COVER CONSTRAINT
C  MUST BE SATISFIED.
C
      S=SAVAIL
      SMNLMT=SC
      RETURN
      END
```

```
      SUBROUTINE VMAXX(QS,VMAX,SIZPIP,S,SMNLMT,IOK,S2)
C
C   QS=MAXIMUM FLOW AT END OF DESIGN PERIOD.
C   VMAXP=THE VEL. IN THE SEWER (PARTIALLY FULL).
C   VFULL=THE FULL FLOW VELOCITY.
C   VMAX=MAXIMUM VEL. ALLOWED. (USER SUPPLIED)
C   VMAX IS COMPARED TO VMAXP. NOTE: VMAXP MAY BE GREATER
C   THAN VFULL DEPENDING ON WHAT VALUE DRATIO (DRATIO=DEPTH OF
C   FLOW/DIA.OF SEWER) TAKES ON.
C   IOK=1 IMPLIES THAT THIS DIA. WILL SATISFY BOTH MINIMUM
C   AND MAXIMUM VELOCITY CONSTRAINTS.
C
      IOK=1
      S2=S
      IMAX=0
111   CONTINUE
      QFULL=35.71*SIZPIP**2.667*S2**0.5
      AREA=3.1416*SIZPIP*SIZPIP/4.0
      VFULL=QFULL/AREA
      QRATIO=QS/QFULL
      IF(QRATIO.GE.0.14) GO TO 140
      DRATIO=6.23*QRATIO-59.23*QRATIO**2.+214.44*
     1                        QRATIO**3.
      GO TO 150
140   CONTINUE
      IF(QRATIO.GT.0.999) GO TO 350
      DRATIO=0.2+0.73*QRATIO-0.04*QRATIO**2.
150   CONTINUE
      VRATIO=0.018+3.47*DRATIO-7.27*DRATIO**2.+9.14*
     1                  DRATIO**3.-4.28*DRATIO**4.
      VMAXP=VFULL*VRATIO
C   VMAXP HAS BEEN FOUND FOR THE GIVEN DRATIO VALUE AND
C   THE GIVEN S VALUE. IF VMAXP  IS GREATER THAN VMAX
C   THEN FOR THIS PIPE DIA A DROP MANHOLE AT THE U/S END
C   MUST BE USED.
C
       IF(VMAXP.LE.VMAX)RETURN
C   MAX. VEL. CONSTRAINT FOR THIS SLOPE HAS BEEN VIOLATED.
C   REDUCE S AND TRY AGAIN. ALSO, A DROP MANHOLE WILL BE PUT IN
C   A DROP MANHOLE WILL BE PUT IN AT THE U/S END.
C
       IMAX=1
       S2=S2-0.1*S
       IF(S2.LT.SMNLMT) GO TO 350
       GO TO 111
350    CONTINUE
       IF(IMAX.LT.1) GO TO 360
C
C   QF<QS (AND VMAXP>VMAX OR S2<SMNLMT) .THIS PIPE CANNOT BE USED.
C
       IOK=0
       RETURN
360    CONTINUE
C
C   QF IS .LE. QS. VMAXP(VFULL) MAY OR MAY NOT BE GREATER THAN
C   VMAX.SINCE THE PIPE SIZE AT THIS GIVEN SLOPE WILL NOT CARRY THE
C   REQUIRED FLOW .INCREASE S AND CHHECK TO SEE IF VFULL.LT.VMAX.

C   (THHE PIPE WILL BE JUST FLOWING FULL).
C
      S=(QS/(35.71*SIZPIP**2.667))**2.0
      S2=S
      VFULL=QS/AREA
      IF(VFULL.LE.VMAX) GO TO 370
C
C  PIPE SIZE IS TOO SMALL. INCREASE PIPE DIA. AND TRY AGAIN.
C
      IOK=0
      RETURN
370   CONTINUE
      IOK=1
      RETURN
      END
```

```
C
C
C
      SURPOUTINE COSTT(EJSI,GUSI,S,S2,IKOUNT,ICK,NPIPFS,IL,XLI,T,
     1                  GDSI,DDISC,DROP,COST,SIZPIP,XMNCST,EDSOLD,DOPT
     1OL)
C
C  FOR NOW,THERE WILL BE NO DROP COSTS.
C
      DIMENSION CDISC(NPIPES),COST(NPIPES)
C
      IF(IKOUNT.EQ.1) GO TO 30
      EDSOLD=EDSI
30    CONTINUE
C
C  IF S2.NE.S THEN A SROP MANHOLE AT THE U/S END WILL BE
C  BE REQUIRED.
C
      EUS2=EUSI
      EDSI=EUSI-S*XLI
      EUSI=EDSI+S2*XLI
      DROP=EUS2-EUSI
C
C  CALCULATE CUT. THE TERM (SIZPIP+1.)/12. ACCOUNTS FOR THE WALL
C  THICKNESS.  0.33 ACCOUNTS FOR CLEARANCE.
C
      USCUT=GUSI-EUSI+SIZPIP+(SIZPIP+1.)/12.+0.33
      DSCUT=GDSI-EDSI+SIZPIP+(SIZPIP+1.)/12.+0.33
      CUT=0.5*(USCUT+DSCUT)
C
C  PIPE COSTS WEPE BASED ON 1971 PRICES.
C
      PIPCST=3.0*XLI*(9.73+0.097*CUT**1.6+0.826*SIZPIP**3.387)
C
C  CALC. MANHOLE COSTS BASED ON 1971 PRICES. MANHOLE CHAMBER COST
C  COST=$303.. MANHOLES ARE AT THE U/S END OF THE LINK. THE TERM 0.5
C  ACCOUNTS FOR CLEARANCE.
C
      HEIGHT=(GUSI-EUSI)+SIZPIP+0.5
      CSTMNH=303.+121.15+37.81*HEIGHT+0.089*HEIGHT**2-
     1                    0.009*HEIGHT**3
      COST(IL)=PIPCST+CSTMNH
C  IF THIS IS THE FIRST DIA. CONSIDERED THEN NO COMPARISON OF COSTS
C  FOR DIFFERENT DIA. CAN BE MADE.
C
      IF(IKOUNT.EQ.1) GO TO 60
C
C  COMPARE COSTS. IF COST(IL)>COST(IL-1) THEN DIA(IL-1) IS THE
C  OPTIMUM DIA.. IN THE CASE WHERE A DIA. DOES NOT SATISFY THE MAX.
C  VEL. CONSTRAINT (IOK=0) THEN SET COST(IL)=COST(IL-1)-1.. THUS
C  THE SEARCH FOR THE OPTIMUM DIA. WILL CONTINUE.
C
      IF(IOK.NE.0) GO TO 108
      COST(IL)=COST(IL-1)-1.0
      RETURN

108   CONTINUE
C
C  COMPARE COSTS.
C
      IF(COST(IL).GT.COST(IL-1)) GO TO 140
      IF(IL.LT.NPIPES) GO TO 59
      IL=IL+1
      EDSOLD=EDSI
      GO TO 140
59    RETURN
60    CONTINUE
C
C
C  IF THE FIRST DIA. CANNOT SATISFY THE MAX. VEL. CONSTRAINT ASSIGN A
C  HIGH COST TO IT.IF IL=NPIPES THIS IS THE ONLY DIA. THAT CAN BE USED
      IF(IOK.EQ.0) COST(IL)=100000000.
      IF(IL.NE.NPIPES) GO TO 139
      XMNCST=COST(IL)
      DOPTOL=DDISC(IL)
      EDSOLD=EDSI
      IOK=2
  139 RETURN
140   CONTINUE
C
C  DIA(IL-1)  IS THE DIA. WHICH RESULTS IN THE OPTIMUM COST FOR
C  THIS LINK.
C
      XMNCST=COST(IL-1)
      DOPTOL=DDISC(IL-1)
      IOK=2
      RETURN
      END
```

```
      SUBROUTINE STACST(EUSI,GUSI,GDSI,QMAX,QMIN,XLI,
     1                  NPIPES,DDISC,VMAX,T,CBYN,W,XMNCOV,EFF,YEARS,
     1RINT,RINF,                          COST,HP,XCOST,EDSPMP,PMPD
     1IA)
C
C
      DIMENSION DDISC(NPIPES),COST(NPIPES)
C
      REFF=RINT-RINF
      IL=IKOUNT=0
10    IL=IL+1
      IF(IL.LE.NPIPES) GO TO 11
      GO TO 22
11    CONTINUE
      SIZPIP=DDISC(IL)
      AREA=3.1416*SIZPIP*SIZPIP/4.0
      V=QMAX/AREA
      V2=QMIN/AREA
      VMIN=CBYN*(SIZPIP/4.0)**0.1667*(T/W)**0.5
      IF(V.GT.VMAX) GO TO 10
      IF(V2.LT.VMIN) GO TO 20
      CUT=XMNCOV+SIZPIP+(1.0+SIZPIP)/12.+0.33
      CSTPIP=3.0*XLI*(9.73+5.397*CUT**1.6+1.026*SIZPIP**2.387)
      FRICHD=XLI*(QMAX/(35.71*SIZPIP**2.657))**2.0
      DLVRHD=(GDSI-XMNCOV)-EUSI
      HTOT=5.0+DLVRHD+FRICHD+5.0
C
C  CALCULATE PUMP,ELECTRICITY AND O&M COSTS.
C
      HP=(62.4*HTOT*QMAX)/(550.0*EFF)
      HPFAC=1.3 +0.7*EXP(-HP/600.)
      HPI=HP*HPFAC
      IF(HPI.LT.1.0) HPI=1.0
      PMPCST=0.57*ALOG10(HPI)+2.35
      COST1=10.**PMPCST
      HPINIT=(62.4*HTOT*(1.25*QMIN))/(550.0*EFF)
      HPEND =(62.4*HTOT*(0.8*QMAX))/(550.0*EFF)
C
C  COST=$0.0125/HP/HR. OR $81.66/HP/YR.
C
      CSTINT=81.65*HPINIT
      CSTEND=81.65*HPEND
      GRAD=(CSTEND-CSTINT)/YEARS
C
C  FIND ANNUAL COSTS. REFF=RINT-RINF.
C
      A=1.0/REFF
      B=(1.0+REFF)**YEARS-1.0
      ANN=GRAD*(A-(YEARS/B))
      ANNCST=CSTINT+ANN
C
C  CALCULATE PRESENT VALUE COST.
C
      PELECT=ANNCST*(B/(REFF*(1.+REFF)**YEARS))
      AOANDM=0.02*COST1
      POANDM=AOANDM*(B/(REFF*(1.+REFF)**YEARS))
      COST(IL)=CSTPIP+COST1+PELECT+POANDM


      IKOUNT=IKOUNT+1
      IF(IKOUNT.EQ.1) GO TO 10
      IF(COST(IL).LT.COST(IL-1)) GO TO 10
      GO TO 22
20    CONTINUE
      IF(IKOUNT.GT.0) GO TO 22
C
C
      XCOST=1000000000.
      PMPDIA=DDISC(IL)
      HPI=0.0
      EDSPMP=GDSI-XMNCOV
      RETURN
22    CONTINUE
      IF(QMAX.LT.1.25) QMAX=1.25
      BB=0.65*ALOG10(0.8*QMAX)+4.72
      CSTQ=10.0**BB
      CSTQ=0.01*CSTQ*(B/(REFF*(1.+REFF)**YEARS))+CSTQ
      XCOST=COST(IL-1)+CSTQ
      PMPDIA=DDISC(IL-1)
      FRICHD=XLI*(QMAX/(35.71*PMPDIA**2.657))**2.0
      HTOT=5.0+DLVRHD+FRICHD+5.0
      HP=(62.4*HTOT*QMAX)/(550.0*EFF)
      HPFAC=1.3+0.7*EXP(-HP/600.)
      HPI=HP*HPFAC
      EDSPMP=GDSI-XMNCOV
      RETURN
      END
```

```
      SUBROUTINE CENCST(Q,NCENP,B,C,N,NCEN,NQ,TYPE,YEARS,PINT,PINF,
     + DEFICT,EFF,NSTAGE,NSTATE,ND,SIGN,DOPT,FOPT,FD,FMAX,
     + DALLOW,QRIV,P,XK,TREACH,T,DS,XINFNT,FO,FUS,IEFF,CPROD,
     + QPLANT,RTN,OPTF,FSOL,DSOL,NW,BPLANT)
C
C  THIS SUBROUTINE CALCULATES THE COST COEFF. FOR THE TRTMT PLANTS.
C
      DIMENSION Q(NQ),C(NQ),B(N),NCENP(NCEN),CPROD(NCEN)
      DIMENSION EFF(ND),DOPT(NSTAGE,NSTATE),FOPT(NSTATE,NSTATE)
      DIMENSION P(NSTAGE),XK(NSTAGE),DEFICT(NSTATE),DSOL(NSTAGE)
      DIMENSION TREACH(NSTAGE),T(NSTAGE),QPLANT(NCEN),FUS(NSTATE,ND)
      DIMENSION XINFNT(NSTAGE),FD(NSTATE),FMAX(NSTATE),IEFF(NSTATE)
      DIMENSION FSOL(NSTAGE),RTN(ND),OPTF(NSTAGE,NSTATE)
      DIMENSION BPLANT(NCEN)
C
      L2=NQ-NCEN
      DO 10 I=1,NCEN
      ISLACK=L2+I
      ICEN=NCENP(I)
C
C THE 1.3464 IS A CONVERSION FACTOR (CFS TO IMGD). THE TRT*MT PLANT
C COSTS ARE IN TERMS OF IMGD. QPLANT IS ONLY USED TO CALC. COSTS IN
C THE DYNAMIC PROGRAMMING ROUTINE.
C
      QPLANT(I)=(TYPE*B(ICEN)-Q(ISLACK))*1.3464
   10 CONTINUE
C
C  CALL DYNAMIC PROGRAMMING ROUTINES.
C
      CALL DYNDP(DEFICT,EFF,NSTAGE,NSTATE,ND,SIGN,DOPT,FOPT,
     +     FD,FMAX,DALLOW,QRIV,P,XK,TREACH,T,QPLANT,DS,XINFNT,
     +     FO,FUS,IEFF,RTN,OPTF,NW,DSOL,FSOL)
C
      DO 15 I=1,NCEN
      ISLACK=L2+I
      BPLANT(I)=TYPE*B(I)*1.3464
      IF(QPLANT(I).GT.0.01) GO TO 20
C
C  THE COST COEFF.FOR THE REDUNDANT TRT*MT PLANTS WILL BE BASED ON
C  THE PLANT CAPACITY AND ONE UNIT LESS THAN THE CAPACITY. THE LOWEST
C  EFFICIENCY WILL BE USED. NOTE// FOR NOW, IF A TRT*MT PLANTS CAPACITY
C  HAS BEEN SET EQUAL TO ZERO THEN A HIGH COST COEFF. WILL BE ASSIGNED.
C
      QPLANT(I)=BPLANT(I)
      CALL RETFUN(QPLANT,NSTAGE,ND,EFF,I,1,RTN)
      RTN1=RTN(1)
      QPLANT(I)=QPLANT(I)-1.00
      CALL RETFUN(QPLANT,NSTAGE,ND,EFF,I,1,RTN)
      RTN2=RTN(1)
      C(ISLACK)=RTN1-RTN2
      IF(C(ISLACK).EQ.0.0) C(ISLACK)=4.0E07
   20 CONTINUE
C
C  CALC. COST COEFF. FOR THE SLACK FLOWS FOR PLANTS WITH A FINITE FLOW.
C  THE OPTIMUM EFF. FOUND FOR THE PLANT IS USED TO CALC. ALL COST COEFF.
C  CPROD IS ONLY USED IN SUBROUTINE SOLVE TO CALC. THE COST OF THE PLANT.
C  A CHECK SHOULD BE MADE TO SEE IF QPLANT(I)=BPLANT(I). IF IT IS
C  THEN BPLANT(I)MUST BE INCREASED OR ELSE C(ISLACK) WILL BE
C  INDEFINITE.
C
      DIFF=BPLANT(I)-QPLANT(I)
      IF(DIFF.GT.1.00) GO TO 18
      BPLANT(I)=QPLANT(I)+1.00
   18 CONTINUE
      DEC=DSOL(I)
      CALL JSTATE(EFF,ND,DEC,K)
      CALL RETFUN(BPLANT,NSTAGE,ND,EFF,I,K,RTN)
      CAPCTY=RTN(K)
      CALL RETFUN(QPLANT,NSTAGE,ND,EFF,I,K,RTN)
      PRODUC=RTN(K)
      C(ISLACK)=(CAPCTY-PRODUC)/(BPLANT(I)-QPLANT(I))
      CPROD(I)=PRODUC/QPLANT(I)
      WRITE(NW,475) I,CPROD(I),PRODUC,QPLANT(I)
  475 FORMAT(3X,*I,CPROD,PRODUC,QPLANT=*,I3,3F12.2)
   15 CONTINUE
      RETURN
      END
```

```
C
C
C
      SUBROUTINE CALCQ(PPBEGI,PPENDI,CMBEGI,CMENDI,PNOUST,PNENDI,
     1                  ACRESI,QAVG,RINFLT,STPMXI,STPMNI)
      STPMNI=(RINFLT+CMBEGI+PNOUSI)*ACRESI+0.5*QAVG*PPBEGI
      STPMXI=(RINFLT+CMENDI+PNENDI)*ACRESI+2.0*QAVG*PPENDI
      RETURN
      END
```

```
C
      SUBROUTINE JSTATE(STATE,NST,ST,JST)
C
C   THE SUBROUTINE DETERMINES WHICH ELEMENT OF AN ARRAY IS
C   REPRESENTED BYY A SPECIFIC VALUE. THE TOLERANCE IS PASSED
C   THRU A LABELLED COMMON BLOCK.
C
      DIMENSION STATE(NST)
      COMMON /ACC/EPS
      JST=0
      DO 10 J=1,NST
      DIFF=ABS(STATE(J)-ST)
      IF(DIFF.GT.EPS) GO TO 10
      JST=J
   10 CONTINUE
      RETURN
      END
```

```
C
      SUBROUTINE DYNOR(DEFICT,EFF,NSTAGE,NSTATE,ND,SIGN,DOPT,FOPT,FD,
     +  FMAX,DALLOW,QRIV,R,XK,TREACH,T,QPLANT,DS,XINFNT,FO,FUS,
     +  IEFF,RTN,OPTF,NW,DSOL,FSOL)
C
      DIMENSION EFF(ND),DOPT(NSTAGE,NSTATE),FOPT(NSTAGE,NSTATE)
      DIMENSION DEFICT(NSTATE),DSOL(NSTAGE),TREACH(NSTAGE)
      DIMENSION T(NSTAGE),QPLANT(NSTAGE),FUS(NSTATE,ND),P(NSTAGE)
      DIMENSION XINFNT(NSTAGE),FD(NSTATE),FMAX(NSTATE),XK(NSTAGE)
      DIMENSION IEFF(NSTATE),FSOL(NSTAGE),RTN(ND),OPTF(NSTAGE,NSTATE)
      EXTERNAL RETFUN,TRANSF
C
      NPRINT=1
C
C   RESET QRIVER=QRIV (IE.U/S FLOW) FOR EACH ITERATION.
      QRIVER=QRIV
C
      CALL DYNAM(DEFICT,EFF,NSTAGE,NSTATE,ND,SIGN,DOPT,FOPT,FD,FMAX,
     +       DALLOW,QRIVER,R,XK,TREACH,T,QPLANT,DS,XINFNT,FO,FUS,
     +       NW,NPRINT,IEFF,RTN,OPTF,TRANSF,RETFUN)
C
      CALL DYNSOL(DEFICT,EFF,NSTAGE,NSTATE,ND,SIGN,DOPT,FOPT,DS,
     +       QRIVER,R,XK,TREACH,T,QPLANT,XINFNT,OPTF,TRANSF,RETFUN,
     +       RTN,FSOL,NW,NPRINT,DSOL,ANS)
C
      RETURN
      END
```

```
C
C
      SUBROUTINE DYNAM(DEFICT,EFF,NSTAGE,NSTATE,ND,SIGN,DOPT,FOPT,FD,
     +  FMAX,DALLOW,CRIVER,R,XK,TREACH,T,QPLANT,DS,XINFNT,FO,FUS,
     +  NW,NPRINT,IEFF,RTN,OPTF,TRANSF,RETFUN)
C
      DIMENSION EFF(ND),DOPT(NSTAGE,NSTATE),FOPT(NSTAGE,NSTATE)
      DIMENSION R(NSTAGE),XK(NSTAGE),DEFICT(NSTATE)
      DIMENSION TREACH(NSTAGE),T(NSTAGE),QPLANT(NSTAGE)
      DIMENSION FUS(NSTATE,ND),FD(NSTATE),FMAX(NSTATE)
      DIMENSION XINFNT(NSTAGE),IEFF(NSTATE),RTN(ND),OPTF(NSTAGE,NSTATE)
      COMMON/ACC/EPS
      EPS=0.001*ABS(DEFICT(NSTATE)-DEFICT(1))
C  INITIALIZE ALL OPTF(NSTAGE,NSTATE,ND)=FO
      DO 5 JJ=1,NSTATE
      OPTF(1,JJ)=FO
5     CONTINUE
C
      DO 10 N=1,NSTAGE
      DO 13 JK=1,NSTATE
C  SET DEFAULT VALUES ON THE FOLLOWING PARAMETERS.
      FMAX(JK)=-SIGN*1.0E10
      DOPT(N,JK)=EFF(1)
      IEFF(JK)=1
      FUS(JK,1)=10000.0
      FOPT(N,JK)=FMAX(1)
13    CONTINUE
      DO 15 KC=1,ND
      CALL RETFUN(QPLANT,NSTAGE,ND,EFF,N,KC,RTN)
15    CONTINUE
C
      DO 20 J=1,NSTATE
      DO 30 K=1,ND
C
      CALL TRANSF(DEFICT,EFF,NSTAGE,NSTATE,ND,CRIVER,DALLOW,R,XK,
     1            NW,NPRINT,DS,TREACH,T,QPLANT,XINFNT,OPTF,FUS,N,J,K,JDS)
C
C  IF CURRENT SOL*N IS INFEAS. SKIP CALCS. IF FOR ONE STATE ALL SOLNS
C  ARE INFEAS. THEN DOPT(N,J)=DOPT(N,JDS)=EFF(1). (ARBITRARILY) AND
C  FOPT(N,J)=FOPT(N,JDS)=-SIGN*1.0E10 .
C
      IF(JDS.EQ.0) GO TO 30
C  IF WE GET THIS FAR THEN DECISION AND SOL*N ARE FEASIBLE.
      FD(JDS)=RTN(K)
C  IF N=1 THEN IGNORE U/S EFECTS.
      IF(N.EQ.1) GO TO 40
C  NOW,GET U/S STATE AND ADD TO FD(JDS)
      FD(JDS)=FD(JDS)+FOPT(N-1,J)
40    CONTINUE
      IF(SIGN*FD(JDS).LE.SIGN*FMAX(JDS)) GO TO 30
C  UPDATE BEST SOL*N.
      FMAX(JDS)=FD(JDS)
      DOPT(N,JDS)=EFF(K)
      IEFF(JDS)=K
30    CONTINUE
20    CONTINUE
C

      DO 33 JDS=1,NSTATE
      FOPT(N,JDS)=FMAX(JDS)
33    CONTINUE
C
C  UPDATE THE F VALUES CORRESPONDING TO THE OPTIMUM DECISION FOR EACH
C  STATE. OPTF WILL BE USED IN THE NEXT REACH.
C  THE VALUE OF (N+1) IN OPTF(N+1,JDS) IS NOT IN ACCORDANCE WITH THE
C  REACH NUMBER.OPTF(N+1,JDS) CORRESPONDS TO THE FUS(JDS,IDEC)-VALUE
C  FOR THE REACH DIRECTLY U/S (IE. REACH(I-1)).
C
      IF(N.EQ.NSTAGE) GO TO 10
      DO 34 JDS=1,NSTATE
      IDEC=IEFF(JDS)
      OPTF(N+1,JDS)=FUS(JDS,IDEC)
34    CONTINUE
C
10    CONTINUE
      RETURN
      END
```

```
C
C
      SUBROUTINE TRANSF(DEFICT,EFF,NSTAGE,NSTATE,ND,QRIVER,DALLOW,R,XK,
     1    NW,NPRINT,DS,TREACH,T,QPLANT,XINFNT,OPTF,FUS,I,J,K,JDS)
C
      DIMENSION EFF(ND),R(NSTAGE),XK(NSTAGE),QPLANT(NSTAGE)
      DIMENSION T(NSTAGE),DEFICT(NSTATE),TREACH(NSTAGE)
      DIMENSION XINFNT(NSTAGE),OPTF(NSTAGE,NSTATE),FUS(NSTATE,ND)
C
      IF(J.GT.1.OR.K.GT.1) GO TO 8
C
      IF(I.GT.1) GO TO 7
      KOUNT=NSTATE
      ICOUNT=1
      QTOT=QRIVER+QPLANT(I)
      GO TO 8
7     CONTINUE
      ICOUNT=KOUNT
      KOUNT=NSTATE
      QRIVER=QTOT
      QTOT=QTOT+QPLANT(I)
8     CONTINUE
C
      IF(J.GE.ICOUNT) GO TO 12
      JDS=0
      RETURN
12    CONTINUE
C
C  DEFICT(J) REPRESENTS THE WATER QUAALITY (MEASURED IN TERMS OF DEFICIT)
C  AT THE END OF REACH(I-1).
C
      DEFTOP=(QRIVER*DEFICT(J)+T(I)*QPLANT(I))/QTOT
      IF(DEFTOP.LE.DALLOW) GO TO 10
      JDS=0
      RETURN
10    CONTINUE
C
C  CALCULATE THE BOD CONC. AT THE TOP OF REACH I. THIS CONC. IS DEPENDEN
C  UPON OPTF(J) AND THE EFFICIENCY OF THE PLANT LOCATED AT THE TOP OF
C  REACH I .OPTF(J) IS THE VALUE OF F=H(I-1)*BODTOP(I-1),WHERE  (I-1)
C  SIGNIFIES THE U/S REACH.THE VALUE OF F (IE.OPTF(J)) USED IS THE
C  CORRESPONDING VALUE FOR THE OPTIMUM DECISION IN REACH (I-1) FOR THE
C  CURRENT VALUE OF J; (IE. E FOR REACH (I-1)).
C
      XMK=(1.00-EFF(K))*XINFNT(I)
      BODTOP=(QRIVER*OPTF(I,J)+XMK*QPLANT(I))/QTOT
C
C  CALCULATE F FOR REACH I . F IS DEPANDENT ON BODTOP AND THUS BOTH
C  OPTF(J) AND XMK.
C
      HIJ=EXP(-XK(I)*TREACH(I))
      F=HIJ*BODTOP
C
C  CALC. TIME TO REACH MAX. DO DEFICT.
      A=1.0/(R(I)-XK(I))
      B=((R(I)-XK(I))*DEFTOP)/(XK(I)*BODTOP)
      C=(R(I)/XK(I))*(1.00-B).

C  IF C.LE.1.00 THEN TC WILL BE .LE. 0.0 .THUS DEFTOP IS THE LARGEST
C  DEFICIT WITHIN THE REACH. IE. TO GET THIS FAR DEFTOP.LE. DALLOW .
      IF(C.LE.1.0C) GO TO 20
      TC=A*ALOG(C)
C
C
C  TEST TC AGAINST TREACH(I). IF TC>TREACH(I) THEN MAX. DEFICIT OCCURS
C  OUTSIDE OF REACH. THUS, FIND ENEW(JDS,K) AND COMPARE TO DALLOW.
C  IF TC<TREACH(I) TEST DC AGAINST DALLOW. CALC. E(JDS,K) IF DC<DALLOW.
C
      IF(TC.GT.TREACH(I)) GO TO 20
C  TC< OR= TREACH(I), CALC. DC AND COMPARE TO DALLOW.
      DC=(XK(I)*BODTOP/R(I))*EXP(-XK(I)*TC)
      IF(DC.LE.DALLOW) GO TO 20
C  DEC. AND SOL+N. ARE INFEAS.
      JDS=0
      RETURN
C
20    CONTINUE
      FA=XK(I)/(R(I)-XK(I))
      FB=EXP(-XK(I)*TREACH(I))-EXP(-R(I)*TREACH(I))
      FIJ=FA*FB
      GIJ=EXP(-R(I)*TREACH(I))
      E=FIJ*BODTOP+GIJ*DEFTOP
      IF(E.LE.DALLOW) GO TO 30
      JDS=0
      RETURN
30    CONTINUE
C  DISCRETIZE E.
      DISCE=FLOAT(IFIX(E/DS+0.5))*DS
      CALL JSTATE(DEFICT,NSTATE,DISCE,JDS)
      FUS(JDS,K)=F
      IF(JDS.LT.KOUNT) KOUNT=JDS
      RETURN
      END
```

```
C
C
      SUBROUTINE RETFUN(QPLANT,NSTAGE,ND,EFF,I,K,RTN)
      DIMENSION QPLANT(NSTAGE),EFF(ND),RTN(ND)
C
      FACTOR=17.29
      CSTTEN=7164121.
      IF(QPLANT(I).GT.0.01) GO TO 5
      RTN(K)=0.0
      RETURN
    5 CONTINUE
      IF(QPLANT(I).LE.1.00) QPLANT(I)=1.00
      A=-0.194*ALOG10(QPLANT(I))+5.79
      COST1=10.0**A*QPLANT(I)
      AA=-0.23*ALOG10(QPLANT(I))+4.5
      COST2=(10.**AA)*FACTOR*QPLANT(I)
C  FACTOR=CONVERSION FACTOR FROM ANNUAL TO PRESENT VALUE.
      TOTCST=COST1+COST2
      FAC=TOTCST/CSTTEN
C  FAC ACCOUNTS FOR ECONOMIES OF SCALE IN BUILDING THE PLANT.
C  NOW,ACCOUNT FOR INCREASING COST WITH INCREASING EFFICIENCY.
C  THE CURVE USED FOR COST VS. EFF. HAS BEEN BROKEN INTO THREE
C  LINEAR PARTS. THE TRTMT EFF. FOR THIS DATA MUST BE 0.35<EFF<0.97 .
C
      IF(EFF(K).GT.0.87) GO TO 10
      RTN(K)=3.48E06*EFF(K)*FAC+3.22E05*FAC
      RETURN
   10 CONTINUE
      IF(EFF(K).GT.0.94) GO TO 20
      RTN(K)=1.237E07*EFF(K)*FAC-4.515E06*FAC
      RETURN
   20 CONTINUE
      RTN(K)=1.546E08*EFF(K)*FAC-1.382E08*FAC
      RETURN
      END
```

```
C
C
      SUBROUTINE DYNSOL(DEFICT,EFF,NSTAGE,NSTATE,ND,SIGN,DOPT,FOPT,DS,
     +  QRIVER,R,XK,TREACH,T,QPLANT,XINFNT,OPTF,TRANSF,RETFUN,
     +  RTN,FSOL,NW,NPRINT,DSOL,ANS)
C
      DIMENSION DOPT(NSTAGE,NSTATE),FSOL(NSTAGE),FOPT(NSTAGE,NSTATE)
      DIMENSION TREACH(NSTAGE),T(NSTAGE),QPLANT(NSTAGE),RTN(ND)
      DIMENSION XINFNT(NSTAGE),OPTF(NSTAGE,NSTATE),DSOL(NSTAGE)
      DIMENSION DEFICT(NSTATE),EFF(ND),R(NSTAGE),XK(NSTAGE)
      COMMON/ACC/EPS
C   FIND MIN.RETURN.
      FMAX=-SIGN*1.0E10
      DO 10 JDS=1,NSTATE
      IF(SIGN*FOPT(NSTAGE,JDS).LT.SIGN*FMAX) GO TO 10
      FMAX=FOPT(NSTAGE,JDS)
      DSOL(NSTAGE)=DOPT(NSTAGE,JDS)
      JOPT=JDS
10    CONTINUE
      ANS=FMAX
      DEC=DSOL(NSTAGE)
      N=NSTAGE
      CALL JSTATE(EFF,ND,DEC,K)
      CALL RETFUN(QPLANT,NSTAGE,ND,EFF,N,K,RTN)
      FSOL(N)=RTN(K)
C   FIND U/S STATE VARIABLE.
      QTOT=QRIVER+QPLANT(NSTAGE)
5     CONTINUE
      CALL JSTATE(EFF,ND,DEC,K)
      CALL TRANS2(DEFICT,EFF,NSTAGE,NSTATE,QRIVER,R,XK,DS,QTOT,
     1            NW,NPRINT,FOPT,TREACH,T,QPLANT,XINFNT,OPTF,N,JOPT,K,JUS)
C   FIND OPTIMUM EFFICIENCY FOR REACH (N-1).
      N=N-1
      DSOL(N)=DOPT(N,JUS)
      DEC=DSOL(N)
      CALL JSTATE(EFF,ND,DEC,K)
      CALL RETFUN(QPLANT,NSTAGE,ND,EFF,N,K,RTN)
      FSOL(N)=RTN(K) .
      JOPT=JUS
      IF(N.EQ.1) GO TO 7
      GO TO 5
7     CONTINUE
C   PRINT OUT TABLE AND SOLN*S.
      NPRINT=1
      IF(NPRINT.EQ.0) RETURN
      WRITE(NW,35)ANS
35    FORMAT(26H    OPTIMUM COST OR RETURN=,E14.5)
      DO 36 I=1,NSTAGE
      WRITE(NW,37)I,DSOL(I),I,FSOL(I)
37    FORMAT(4X,*XOPT(*,I2,*)=*,E14.5,4X,*FSOL(*,I2,*)=*,E14.5)
36    CONTINUE
      NPRINT=0
      IF(NPRINT.EQ.0)RETURN
C   THE ABOVE TWO LINES SHOULD BE TAKEN OUT WHEN FIXING UP THE PROGRAM
      DO 100 I=1,NSTAGE
      WRITE(NW,101) I
101   FORMAT(//,12H    AT STAGE,I4,/,

     1                                             10X,5HSTATE,10X,4HDOPT,
     110X,4HFOPT)
      DO 110 JDS=1,NSTATE
      WRITE(NW,111)DEFICT(JDS),DOPT(I,JDS),FOPT(I,JDS)
111   FORMAT(3X,3E14.5)
110   CONTINUE
100   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE TRANS2(DEFICT,EFF,NSTAGE,NSTATE,QRIVER,R,XK,DS,
     1    QTOT,NW,NPRINT,FOPT,TREACH,T,QPLANT,XINFNT,OPTF,I,JOPT,K,JUS)
C
      DIMENSION DEFICT(NSTATE),EFF(NSTATE),R(NSTAGE),XK(NSTAGE)
      DIMENSION TREACH(NSTAGE),T(NSTAGE),QPLANT(NSTAGE)
      DIMENSION XINFNT(NSTAGE),FOPT(NSTAGE,NSTATE),OPTF(NSTAGE,NSTATE)
      COMMON/ACC/EPS
C
C   PURPOSE OF THIS SUBROUTINE IS ONLY TO FIND THE VALUE OF J WHICH
C   LEADS TO THE OPTIMUM STATE.
C
      IF(I.EQ.NSTAGE) GO TO 10
      QTOT=QTOT-QPLANT(I+1)
      QRIVER=QTOT-QPLANT(I)
10    CONTINUE
      J=0
      DISCE=DEFICT(JOPT)
20    J=J+1
      IF(J.EQ.NSTATE) GO TO 30
C   ONLYY CHECK FEASIBLE WATER QUALITYY STATES FOR THE PREVIOUS
C   (IE. THE U/S) REACH.
      IF(FOPT(I-1,J).EQ.1.0E10) GO TO 20
      DEFTOP=(QRIVER*DEFICT(J)+T(I)*QPLANT(I))/QTOT
      XMK=(1.00-EFF(K))*XINFNT(I)
      BODTOP=(QRIVER*OPTF(I,J)+XMK*QPLANT(I))/QTOT
      FA=XK(I)/(R(I)-XK(I))
      FB=EXP(-XK(I)*TREACH(I))-EXP(-R(I)*TREACH(I))
      FIJ=FA*FB
      GIJ=EXP(-R(I)*TREACH(I))
      ECOMP=FIJ*BODTOP+GIJ*DEFTOP
      DECOMP=FLOAT(IFIX(ECOMP/DS+0.5))*DS
      DIFF=ABS(DECOMP-DISCE)
      IF(DIFF.LE.EPS) GO TO 30
      GO TO 20
30    CONTINUE
      JUS=J
      RETURN
      END
```

APPENDIX F

INPUT DATA

Part of the data which is required for the program is stored in a data file and read in by subroutine DEFINE. The remainder of the data is entered directly in the main program.

The following information should be included in the data file. The required formats are shown in brackets.

NO. OF LINES OF DATA	DATA
1	Line number 1 contains the number of nodes in the network, N, the number of potential treatment facilities, NCEN, the number of transportation links, L, and a variable, TYPE, which is set equal to -1.0. FORMAT (3I5,F10.1).
2 x N	Line number 2 contains a node number, I, the population at the beginning of the design period, POPBEG(I), the population at the end of the design period, POPEND(I), the commercial flow at the beginning of the design

period, COMBEG(I), and the commercial flow at the end of the design period, COMEND(I). Flows are in cfs. FORMAT(I5,4F10.0).

Line 3 contains the same node number, I, the industrial flow at the beginning of the design period, RINDUS(I), the industrial flow at the end of the design period, RINEND(I), and the incremental acreage for this node, ACRES(I). If the node is a treatment facility then all flowrates and the incremental acreage are set equal to 0.00. Two lines are required for each node. Flows are in cfs. FORMAT(I5,3F10.0).

N For each node the following data is required. The node number, I, identification name for the node and the ground elevation for the node, STATE(I). FORMAT(I3,3X,4A5,F10,1).

L For each transportation link the following data is required. The upstream node number of the link, NUS(I), the downstream node number of the link, NDS(I), the link length in one direction, XL(I2), the link length in the opposite direction, XL(I2P1) the initial flow in one direction, Q(I2), and the initial flow in the opposite direction, Q(I2P1). Generally the link lengths will be identical. If they are identical then the link length in the second flow direction may be left blank. In addition, if the initial flowrates are to be the same for all transportation links then the initial flowrate may be defined in the main program by the variable FLOW. The last two entries of each line would therefore be left empty. FORMAT(2I5,3X,4F10.2).

The following data is entered directly in the main program:

(1) The minimum pipe cover in feet, XMNCOV.

(2) The number of possible pipe diameters for each link, NPIPES.

(3) The minimum allowable pipe diameter in feet, DMIN.

(4) The average daily per capita supply waste flow expressed in cfs., QAVG. Typically QAVG = 105.0 I mgd or 0.000197 cfs.

(5) The average infiltration rate per day per acre, expressed in cfs/acre-day, RINFLT.

(6) An array defining the set of possible pipe diameters in feet, DDISC.

(7) An initial flowrate for all transportation links in cfs, FLOW. The user may instead assign different initial flowrates to different links. In this case

FLOW would be set equal to 0.00.

(8) The number of discrete levels of oxygen deficits, NSTATE.

(9) The maximum allowable oxygen deficit in mg/l, DALLOW.

(10) The number of possible treatment efficiencies, ND.

(11) A variable, SIGN, set equal to -1.0.

(12) The oxygen deficit increment, DS. DS is equal to DALLOW/(NSTATE-1).

(13) The BOD concentration just upstream of the furthermost upstream reach, in mg/l, FO.

(14) The river flowrate in Imgd, QRIV.

(15) An array defining the set of possible treatment efficiencies, EFF.

(16) An array defining the set of possible discrete levels of oxygen deficits, DEFICT.

(17) An array defining the BOD concentration at the inlet point to each plant in mg/l, XINFNT.

(18) An array defining the deoxygenation coefficients for each reach, R.

(19) An array defining the deoxygenation coefficients for each reach, XK.

(20) An array defining the time of travel in days for each reach, TREACH.

(21) An array defining the oxygen deficit in mg/l for the treated waste at each plant, T.

LISTING OF INPUT DATA

40	4 66	-1.0		
1	0.00	0.00	0.00	0.00
1	0.00	0.00	0.00	
2				
2				
3				
3				
4				
4				
5	1400.0	1400.0		
5			1.00	
6	1960.0	1960.0		
6			1.00	
7	1970.0	1970.0		
7			1.00	
8	3000.0	3000.0		
8			1.00	
9	3000.0	3000.0		
9			1.00	
10	1500.0	1500.0		
10			1.00	
11	14000.0	14000.0		
11			1.00	
12	6000.0	6000.0		
12			1.00	
13	15000.0	16000.0		
13			1.00	
14	7000.0	7000.0		
14			1.00	
15	7000.0	7000.0		
15			1.00	
16	2000.0	2000.0		
16			1.00	
17	[illegible]00.0	4000.0		
17			1.00	
18	6000.0	6000.0		
18			1.00	
19	20675.0	20675.0		
19			1.00	
20	8000.0	8000.0		
20			1.00	
21	15000.0	15000.0		
21			1.00	
22	4000.0	4000.0		
22			1.00	
23	3771.0	3771.0		
23			1.00	
24	7614.0	7614.0		
24			1.00	
25	20150.0	20150.0		
25			1.00	
26	20150.0	20150.0		
26			1.00	
27	2000.0	2000.0		
27			1.00	
28	1500.0	1500.0		
28			1.00	
29	3000.0	3000.0		
29			1.00	
30	3990.0	3990.0		
30			1.00	
31	8000.0	8000.0		
31			1.00	
32	1000.0	1000.0		
32			1.00	

```
 33    7000.0     7000.0
 33                           1.00
 34    1000.0     1000.0
 34                           1.00
 35    7000.0     7000.0
 35                           1.00
 36   10000.0    10000.0
 36                           1.00
 37    6000.0     6000.0
 37                           1.00
 38    4420.0     4420.0
 38                           1.00
 39    4000.0     4000.0
 39                           1.00
 40    5000.0     5000.0
 40                           1.00
 3    PROCESSING NODE 3      960.0
 2    PROCESSING NODE 2     1000.0
 1    PROCESSING NODE 1     1050.0
 4    PROCESSING NODE 4      880.0
 5    NODE 5                 985.0
 6    NODE 6                 942.0
 7    NODE 7                 951.0
 8    NODE 8                 961.0
 9    NODE 9                 973.0
10    NODE 10                985.0
11    NODE 11                968.0
12    NODE 12                972.0
13    NODE 13                978.0
14    NODE 14                986.0
15    NODE 15                960.0
16    NODE 16                971.0
17    NODE 17                965.0
18    NODE 18                981.0
19    NODE 19                976.0
20    NODE 20                986.0
21    NODE 21                988.0
22    NODE 22                998.0
23    NODE 23               1010.0
24    NODE 24               1025.0
25    NODE 25               1015.0
26    NODE 26               1039.0
27    NODE 27                993.0
28    NODE 28                995.0
29    NODE 29                971.0
30    NODE 30                968.0
31    NODE 31                978.0
32    NODE 32                951.0
33    NODE 33                958.0
34    NODE 34                935.0
35    NODE 35                911.0
36    NODE 36                933.0
37    NODE 37                921.0
38    NODE 38                895.0
39    NODE 39                903.0
40    NODE 40                892.0
  5     3    15500.00
  6     7     6900.00
  7     3    14600.00
  7     9    23000.0
 10     2    11000.0
  9    10     6100.0
  8     9     6000.0
  9    13     7000.0
 11     9     4800.0
```

9	13	2300.0
11	12	2000.0
10	14	6500.0
13	14	2700.0
12	14	5000.
12	13	3200.
14	2	11000.
18	2	8300.
[illegible]	19	5000.
[illegible]3	15	5300.
15	16	4500.
15	17	4500.
17	18	7100.
17	19	6000.
19	18	6500.
20	18	5900.
19	20	3200.
20	2	11000.
22	20	10100.
21	22	5100.
22	23	6500.
23	24	5900.
24	1	10700.
21	24	14500.
25	1	12600.
25	1	6500.
25	26	6500.
22	25	16100.
31	21	9000.
28	25	10000.
30	31	6300.
30	33	6600.
33	31	6600.
29	30	5500.
29	29	12500.
27	28	11700.
28	5	12000.
27	29	15800.
27	5	6400.
5	29	11000.
29	32	7700.
32	38	9300.
33	34	13100.
32	34	11500.
34	35	33500.
36	15	30000.
37	36	6500.
15	39	37100.
39	5	32100.0
35	37	10100.
35	39	9100.
37	38	12300.
37	39	13200.
38	4	8000.
39	4	3500.
39	40	5200.
40	4	5900.

APPENDIX G

COST DATA

This section outlines the costs used in this study. In addition, a detailed description of the relevant factors which should be considered when selecting a pumping station is given. The cost equations which are presented are intended to be used only as a guideline. The user may easily modify the existing equations or supply his own equations. Any additional parameters which are added into the cost equations must be transferred through the relevant calling statements.

Link Costs for Gravitational Flow

The costs for a link include the cost for a manhole, and the cost for the pipe and required excavation. The costs were taken from Main [21]. The quoted costs include all items of work except shoring of excavations, restoring existing improvements, and restoring existing permanent surfacing.

Manhole Costs:

$$CSTMNH = 424.18 + 37.81 \times HEIGHT + 0.089 \times (HEIGHT)^2 - 0.009 \times (HEIGHT)^3 \quad (G.1)$$

where,

CSTMNH = cost of manhole (\$)

HEIGHT = height of shaft (ft)

Pipe and Excavation Costs:

$$PIPCST = 3.0 \times XLI \times (9.73 + 0.097 \times (CUT)^{1.6} + 0.826 \times (SIZPIP)^{3.387}) \quad (G.2)$$

where,

PIPCST = cost for excavation and pipe (\$)

XLI = link length (ft)

SIZPIP = diameter of pipe (ft)

CUT = average depth to subgrade (ft)

Cost Data for the Pumping Station and Force Main

The construction cost for raw wastewater pumping stations may be siginficantly affected by a number of factors. These factors include; the pumping capacity and type of pumps to be used, influent sewer depth, pumping head, provision for growth, the number of pumps, and the location and external appearance of the pumping station.

It would be extremely difficult to incorporate all these factors into a design model. Therefore, only the most critical factors have been considered here.

Before one can choose the relevant factors involved in designing a pumping station a general type of station, including the type of pump to be used, should be determined. Bartlett [4] and MOP9 [2] provided useful guidelines for answering these questions.

The most common type of pump used for raw sanitary sewage is the centrifugal pump. A non-clog pump which is capable of passing solids up to 100 mm (4 in.) in size should be used. Vertical spindle pumps with a cast iron casing are common and economical. Non-clog pumps generally operate at an efficiency of 50-60 percent.

The cost of a pumping scheme as opposed to that of a gravity scheme should include both capital costs as well as estimated running costs. The estimated running costs consist of electricity costs and operation and maintenance costs.

Figure G.1 schematically represents a pumping station and force main. Some pumping stations discharge directly into open channels. The majority, however, pump the sewage through a force main for a considerable distance. The pumping station will be at the upstream end of a link. The force main will discharge the sewage to the delivery manhole

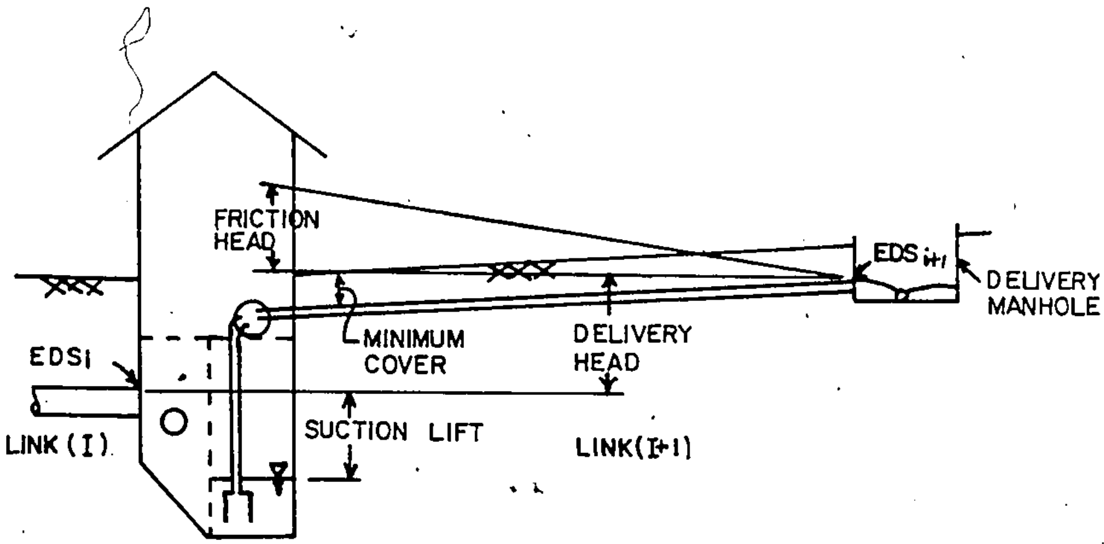


FIGURE G-1 - FORCE MAIN AND PUMPING STATION SCHEME at atmospheric pressure.

The pumping station and force main, together, will comprise one link in the system.

The total head which must be delivered is:

$$\text{TOTAL HEAD} = \text{SUCTION LIFT} + \text{DELIVERY HEAD} + \text{FRICTION LOSSES} + \text{MINOR LOSSES} \quad \text{(G.3)}$$

The suction lift varies with variations in the water level in the wet well. The suction lift will be at its maximum when the maximum output is required. Typical design dimensions from Bartlett [4] show this value to be equal to 5 feet. Minor losses occur due to fittings and valves. These losses may be accounted for with one value, typically 4 to 5 feet.

The delivery head is defined as $(EDS_{i+1} - EDS_i)$.

The friction losses may be represented by the Manning's equation:

$$S = \left(\frac{n \times Q}{1.49 \times A \times R^{2/3}}\right)^2 = \left(\frac{n \times Q}{0.46 \times d^{8/3}}\right)^2 \quad \text{(G.4)}$$

Therefore, the total head (HTOT) is defined as:

$$HTOT = 5' + (EDS_{i+1} - EDS_i) + \text{length} \times \left(\frac{n \times Q}{0.46 \times d^{8/3}}\right)^2 + 5' \quad \text{(G.5)}$$

where

length = length of force main (ft).

For sanitary sewage pumping, the pumps must be capable of pumping, at the required total head, the peak flows as they occur in the sewers. The maximum flow at the end of the design period will be used as the peak flow. The total pumping capacity should be such that the maximum flow can be handled with one unit out of service. Thus, even for small pumping stations two pumps will be required.

If two units are installed, each one should have the required horsepower to handle the maximum flow at the required head. If three units were installed then each unit should have the required horsepower to handle one half of the maximum flow, at the required total head.

Generally, as the total horsepower for a pumping station increases the size of the units will increase, or the number of units will increase. To account for the extra horsepower which must be installed, due to the possiblity of one unit being out of service, a factor, HPFAC, has been calculated. This factor accounts for the necessary extra installed horsepower. The value of HPFAC may be estimated using the following equation:

$$HPFAC = 1.3 + 0.7\ e^{(-HP/600)} \qquad (G.6)$$

The equation shows that, as the total required horsepower increases the extra installed capacity will decrease.

The equation for the calculation of horsepower is;

$$HP = \frac{62.4 \times HTOT \times Q}{550.0 \times \mu} \qquad (G.7a)$$

where,

μ = efficiency.

The installed horsepower will then be:

$$HPI = HP \times HPFAC \qquad (G.7b)$$

An equation relating pump costs to horsepower was found using data supplied by Woods*. The pumps were centrifugal, vertical spindle, with cast iron casings. The costs include delivery changes. The equation is

$$\log(COST1) = 0.57 \log(HPI) + 2.35 \qquad (G.8)$$

let

$$PMPCST = 0.57 \log (HPI) + 2.35 \qquad (G.9)$$

therefore

$$COST1 = 10^{PMPCST} \qquad (G.10)$$

*Woods, Dr. E.R. PUMP COSTS. Private communication from Dr. E.R. Woods, 1979, Department of Chemical Engineering, McMaster University, Hamilton, Ontario.

Table G.1 shows the costs of the pumps for selected values of installed horsepower.

Associated with the pump costs will be electrical costs. The electrical costs should be based on the average, not the maximum flowrates.

The method used to calculate the electrical costs is as follows. The required horsepower, based on the average flow at the beginning of the design period was calculated. The required horsepower at the end of the design period was also calculated. The electricity cost was then assumed to increase uniformly between these two periods. Using this method it is possible to represent the total electricity costs in the following manner:

1. Find the initial electricity costs based on the initial required horsepower.
2. Find the final electricity costs based on the final required horsepower.
3. Calculate the gradient:

   GRAD = (COST END - COST INIT)/DESIGN PERIOD (G.11)

4. Find the total annual costs, where the total annual cost, ANNCST = COST INIT + ANNUAL GRADIENT COST. The annual gradient cost (ANN) is equal to

$$ANN = GRAD^{*}\left(\frac{1}{REFF} - \frac{N}{(1+REFF)^{N}-1}\right) \qquad (G.12)$$

where,

N = design period

REFF = interest rate minus inflation rate.

5. This annual electricity cost can then be converted to a present value cost.

The pump station costs are based on the average flow at the end of the design period. Generally, the costs are only accurate within a certain range of total head. The pumping stations will normally remain within this range of head due to significant initial costs involved for the pumps and pumping station.

An equation relating the pumping station cost to flow was found using data supplied by Woods.

The pumping station cost includes normal earthwork, the pumping station structure, drives, controls and integral piping. The equation is:

$$\log(COST2) = 0.65 \log(Q) + 4.72 \quad (G.13)$$

let

$$STACST = 0.65 \log(Q) + 4.72 \quad (G.14)$$

therefore

$$COST2 = 10^{STACST} \quad (G.15)$$

If Q is less tha 1.0 cfs then STACST = 4.72.

Table G.2 shows the pump station costs for selected values of flow.

The annual cost due to operation and maintenance (AOANDM) may be taken to be a percentage of the pump and pumping station costs. Thus,

$$AOANDM = K_1\ (COST1) + K_2(COST2) \qquad (G.16)$$

where,

K_1 = may be taken to be 0.02. The value is assumed arbitrarily to allow for earlier replacement of pumps. If desired a sinking fund cost could be worked out once a schedule of pump installations has been determined, and

K_2 = 0.01.

The pipe cost for a force main depends on the type of material used. Materials used today include steel (protected inside and out with bitumen), asbestos-cement, ductile iron and plastics.

The cost per lineal foot, for a steel pipe and associated excavation costs has been taken from Main [21] and modified to give:

$$CSTPIP = 3.0 \times XLI \times (9.73 + 0.097 \times (CUT)^{1.6} + 1.026 \times (DIA)^{2.387}) \qquad (G.17)$$

where,

CSTPIP = cost for excavation and pipe (dollars).

This cost equation is not consistent with the cost

TABLE G-1 - PUMPING COST vs. INSTALLED HORSEPOWER

Pump Cost $	Installed Horsepower
560.	5
831.	10
1402.	25
2081.	50
3090.	100

TABLE G-2 - PUMPING STATION COST vs. FLOW

Pump Station Cost ($)	Flow (cfs)
52,480.	1.0
95,200.	2.5
149,400.	5.0
234,400.	10.0
425,300.	25.0

* Costs have been rounded to 4 significant figures.

equation used for gravity sewers (see equation G.2). For large diameters the cost for a force main will be significantly less than for that of a gravity sewer of equal length, diameter and excavation depth. This inconsistency has resulted in final solutions which will, in all likelihood, contain an excess number of pumping stations and force mains.

The total cost for a pumping station and the associated force main therefore, consists of pump costs, electricity costs, pumping station costs, annual operation and maintenance costs as well as the force main and excavation costs.

Treatment Plant Costs

A simplified method was used to calculate the treatment plant costs in this study. A study by R. Smith [21] was used to derive a cost vs. flow equation. However, no specific treatment efficiency was quoted in Smith's study. An efficiency of 85 percent was assumed. Two equations are used to define the cost vs. flow cost. The first equation accounts for the capital costs, the second equation accounts for annual operation and maintenance costs.

Capital Costs for a Treatment Plant:

$$A = -0.194 \times \log(Q) + 5.79 \qquad \text{(G.18)}$$

$$COST1 = 10^{A} \times Q \quad (G.19)$$

where,

Q = treatment plant flow (Imgd)

Operation and Maintenance Costs for a Treatment Plant:

$$AA = -0.23 \times \log(Q) + 4.5 \quad (G.20)$$

$$COST2 = 10^{AA} \times FACTOR \times Q \quad (G.21)$$

where,

FACTOR = conversion factor, converting annual costs to present value costs.

Tables G.3 and G.4 show capital and annual costs respectively for several flow values. The annual cost was converted to a present value cost.

The cost vs. treatment efficiency (% BOD removal) equation was derived from a cost curve from Rossman and Liebman [24]. The cost vs. efficiency curve was quoted only for a flow of 10.0 Imgd. It was therefore necessary to develop cost vs. efficiency curves for other plant flows. A term, FAC, was used to derive these additional curves. FAC is equal to the total cost of a treatment plant for a flow Q and treatment efficiency of 85 percent divided by the total cost for a 10.0 Imgd plant and treatment efficiency of 85 percent. The cost curves were then linearized and are stated as follows:

TREATMENT EFF., e; $0.35 \leq e \leq 0.87$

$$RTN = (3.48 \times 10^6 \times e + 3.22 \times 10^6) \times FAC \qquad (G.22)$$

TREATMENT EFF., e; $0.87 < e \leq 0.94$

$$RTN = (1.237 \times 10^7 \times e - 4.515 \times 10^6) \times FAC \qquad (G.23)$$

TREATMENT EFF., e; $0.94 < e$

$$RTN = (1.546 \times 10^8 \times e - 1.382 \times 10^8) \times FAC \qquad (G.24)$$

Table G.5 lists several costs for various efficiencies (Q = 10.0 Imgd).

This method assumes that any two cost vs. efficiency curves for two flowrates will be separated by a factor, FAC, which is equal to the associated economies of scale in building the plants.

A more accurate method for determining the set of cost vs. efficiency curves was used by Graves, Hatfield and Whinston [13]. The cost vs. efficiency curves were found directly for several different flowrates. Linear interpolation could then be used to calculate costs for intermediate flowrates.

TABLE G-3 - TREATMENT PLANT CAPITAL COST vs. FLOW

Capital Cost ($)	Plant Flow (Imgd)
616,600	1
3,945,000	10
14,430,000	50
25,230,000	100

TABLE G-4 - TREATMENT PLANT ANNUAL COST vs. FLOW

Annual Cost ($)	Plant Flow (Imgd)
400,000	1
2,356,000	10
7,940,000	50
13,310,000	100

*Costs have been rounded to 4 significant figures.

TABLE G-5 - TREATMENT PLANT COST vs. TREATMENT EFFICIENCY

Plant Cost ($)	% BOD Removal
4,960,000.	50
5,482,000.	65
6,004,000.	80
6,008,000.	85
6,618,000.	90
8,670,000.	95

* Costs have been rounded to 4 significant figures.

5

APPENDIX H

SAMPLE OUTPUT

FOR ITERATION # _ THE FOLLOWING
LINKS COMPRISED THE BRANCHING NETWORK.

					FLOW (CFS)	DIA (FT)	PUMP HP (FT)	U/S INVERT (FT)	D/S INVERT	COST OF LINK $
FROM	NODES	5	TO	[illegible]	5.00	1.50	15.19	934.00	952.00	[illegible].09
FROM	NODES	6	TO	5	5.00	1.50	34.72	952.00	934.00	935.[illegible]
FROM	NODES	3	TO	7	5.00	1.50	69.85	954.00	943.00	3[illegible]0.60
FROM	NODES	7	TO	5	5.00	1.50	0.00	943.00	927.47	390.[illegible]
FROM	NODES	7	TO	7	5.00	1.50	100.90	943.00	952.00	951.76
FROM	NODES	6	TO	6	5.00	1.50	68.90	952.00	948.00	922.09
FROM	NODES	7	TO	9	5.00	1.50	104.80	943.00	965.00	1414.25
FROM	NODES	9	TO	7	5.00	1.50	90.01	965.00	943.00	1350.40
FROM	NODES	10	TO	2	5.00	1.50	99.70	977.00	992.00	772.10
FROM	NODES	2	TO	10	5.00	1.75	0.00	992.00	970.20	1041.09
FROM	NODES	9	TO	10	5.00	1.50	0.00	965.00	951.25	504.07
FROM	NODES	10	TO	9	5.00	1.50	0.00	977.00	965.25	349.04
FROM	NODES	6	TO	9	5.00	1.50	0.00	959.00	959.47	194.55
FROM	NODES	9	TO	6	5.00	1.50	0.00	965.00	951.47	313.49
FROM	NODES	8	TO	13	5.00	1.50	60.00	955.00	970.00	505.[illegible]
FROM	NODES	13	TO	8	5.00	1.50	0.00	970.00	955.00	355.21
FROM	NODES	11	TO	9	5.00	1.50	0.00	963.00	949.18	329.06
FROM	NODES	9	TO	11	5.00	1.50	0.00	965.00	954.18	271.97
FROM	NODES	9	TO	13	5.00	1.50	0.00	965.00	959.81	114.13
FROM	NODES	13	TO	9	5.00	1.50	0.00	970.00	964.81	117.05
FROM	NODES	14	TO	12	5.00	1.50	0.00	963.00	955.49	119.78
FROM	NODES	12	TO	11	5.00	1.50	0.00	964.00	959.19	103.82
FROM	NODES	11	TO	14	5.00	1.50	0.00	977.00	962.04	147.70
FROM	NODES	14	TO	14	5.00	1.50	0.00	978.00	963.34	147.93
FROM	NODES	13	TO	15	5.00	1.50	0.00	970.00	963.91	129.71
FROM	NODES	14	TO	13	5.00	1.50	0.00	970.00	970.00	157.50
FROM	NODES	12	TO	14	5.00	1.50	0.00	964.00	952.73	243.95
FROM	NODES	14	TO	12	5.00	1.50	0.00	978.00	964.00	229.50
FROM	NODES	12	TO	13	5.00	1.50	0.00	964.00	956.73	229.50
FROM	NODES	13	TO	12	5.00	1.50	0.00	970.00	962.73	160.50
FROM	NODES	14	TO	2	5.00	1.50	97.03	978.00	992.00	755.50
FROM	NODES	2	TO	14	5.00	1.75	0.00	992.00	976.28	1005.40
FROM	NODES	16	TO	2	5.00	1.50	95.71	973.00	992.00	606.63
FROM	NODES	2	TO	16	5.00	1.50	0.00	992.00	973.00	1421.[illegible]
FROM	NODES	16	TO	18	5.00	1.50	0.00	963.00	951.75	305.25
FROM	NODES	18	TO	16	5.00	1.50	0.00	973.00	961.75	399.98
FROM	NODES	6	TO	16	5.00	1.50	0.00	953.00	941.05	400.01
FROM	NODES	16	TO	6	5.00	1.50	0.00	963.00	951.05	279.02
FROM	NODES	15	TO	16	5.00	1.50	0.00	952.00	941.83	371.[illegible]
FROM	NODES	16	TO	15	5.00	1.50	0.00	963.00	952.00	220.64
FROM	NODES	15	TO	17	5.00	1.50	0.00	952.00	941.00	356.24
FROM	NODES	17	TO	15	5.00	1.50	0.00	957.00	940.61	355.44
FROM	NODES	17	TO	18	5.00	1.50	64.54	957.00	973.00	500.08
FROM	NODES	18	TO	17	5.00	1.50	0.00	973.00	956.99	500.00
FROM	NODES	17	TO	19	5.00	1.50	0.00	957.00	943.47	105.49
FROM	NODES	19	TO	17	5.00	1.50	0.00	968.00	954.47	319.23
FROM	NODES	19	TO	18	5.00	1.50	0.00	968.00	953.34	479.97
FROM	NODES	18	TO	19	5.00	1.50	0.00	973.00	958.34	396.15
FROM	NODES	20	TO	18	5.00	1.50	0.00	978.00	964.70	350.40
FROM	NODES	18	TO	20	5.00	1.50	0.00	973.00	959.70	424.68
FROM	NODES	19	TO	20	5.00	1.50	0.00	968.00	960.78	243.98
FROM	NODES	20	TO	19	5.00	1.50	0.00	978.00	968.00	162.32
FROM	NODES	20	TO	2	5.00	1.50	97.03	978.00	992.00	775.34
FROM	NODES	2	TO	20	5.00	1.75	0.00	992.00	975.23	1055.40
FROM	NODES	22	TO	20	5.00	1.75	0.00	990.00	975.50	609.08
FROM	NODES	20	TO	22	5.00	1.50	69.96	978.00	990.00	719.99

FROM	NODES	[illegible]	TO	22	5.00	1.50	0.00	980.00	950.50	390.00
FROM	NODES	22	TO	[illegible]	5.00	1.50	0.00	990.00	973.50	100.18
FROM	NODES	22	TO	23	5.00	1.50	74.00	990.00	[illegible]	381.05
FROM	NODES	23	TO	22	5.00	1.50	[illegible]	1002.00	987.04	340.30
FROM	NODES	23	TO	24	5.00	1.50	77.25	1002.00	1027.00	[illegible]
FROM	NODES	24	TO	23	5.00	1.50	[illegible]	1027.00	[illegible]	199.52
FROM	NODES	24	TO	[illegible]	5.00	1.50	127.54	1017.00	1042.00	772.49
FROM	NODES	1	TO	24	5.00	1.50	0.00	1042.00	1027.00 [illegible]	[illegible]
FROM	NODES	21	TO	24	5.00	1.50	157.00	980.00	1017.00	391.52
FROM	NODES	24	TO	21	5.00	1.50	0.00	1017.00	980.00	724.94
FROM	NODES	25	TO	[illegible]	5.00	1.50	148.25	1008.00	1042.00	880.88
FROM	NODES	1	TO	25	5.00	1.50	0.00	1042.00	1018.00 [illegible]	808.74
FROM	NODES	25	TO	1	5.00	1.50	72.02	1031.00	1042.00	529.00
FROM	NODES	1	TO	26	5.00	1.50	0.00	1042.00	1047.00 [illegible]	358.15
FROM	NODES	25	TO	26	5.00	1.50	95.60	1008.00	1001.00	[illegible]
FROM	NODES	26	TO	25	5.00	1.50	0.00	1031.00	1013.00	229.90
FROM	NODES	22	TO	25	5.00	1.50	127.78	990.00	1008.00	1475.13
FROM	NODES	25	TO	22	5.00	1.50	57.41	1030.00	990.00	930.99
FROM	NODES	31	TO	21	5.00	1.50	51.17	970.00	951.00	850.90
FROM	NODES	21	TO	31	5.00	1.75	0.00	930.00	987.12	346.74
FROM	NODES	25	TO	23	5.00	1.50	167.30	987.00	1008.00	729.24
FROM	NODES	25	TO	30	5.00	1.50	0.00	1003.00	985.45	321.00
FROM	NODES	30	TO	31	5.00	1.75	0.00	900.00	951.00	300.53
FROM	NODES	31	TO	33	5.00	1.50	0.00	970.00	955.80	345.08
FROM	NODES	30	TO	33	5.00	1.50	0.00	950.00	945.42	307.00
FROM	NODES	33	TO	[illegible]	5.00	1.50	73.49	950.00	980.00	332.67
FROM	NODES	33	TO	[illegible]	5.00	1.50	90.13	950.00	970.00	540.54
FROM	NODES	31	TO	[illegible]	5.00	1.50	0.00	970.00	950.00	384.90
FROM	NODES	29	TO	[illegible]	5.00	1.50	0.00	903.00	948.04	414.74
FROM	NODES	30	TO	[illegible]	5.00	1.50	0.00	960.00	945.04	402.10
FROM	NODES	23	TO	[illegible]	5.00	1.50	0.00	987.00	952.82	655.22
FROM	NODES	29	TO	23	5.00	1.50	123.72	963.00	987.00	365.51
FROM	NODES	27	TO	28	5.00	1.50	77.40	985.00	987.00	787.97
FROM	NODES	28	TO	27	5.00	1.50	69.50	987.00	985.00	781.49
FROM	NODES	28	TO	5	5.00	1.75	0.00	987.00	969.35	784.34
FROM	NODES	5	TO	28	5.00	1.50	94.43	977.00	987.00	810.08
FROM	NODES	27	TO	29	5.00	1.75	0.00	985.00	981.99	1105.44
FROM	NODES	29	TO	27	5.00	1.50	183.44	965.00	985.00	1300.06
FROM	NODES	27	TO	5	5.00	1.50	0.00	985.00	970.57	366.59
FROM	NODES	5	TO	27	5.00	1.50	0.00	977.00	962.57	497.95
FROM	NODES	5	TO	29	5.00	1.75	0.00	977.00	961.28	955.43
FROM	NODES	29	TO	5	5.00	1.50	97.03	963.00	977.00	770.54
FROM	NODES	29	TO	32	5.00	1.50	0.00	963.00	943.00	390.65
FROM	NODES	32	TO	29	5.00	1.50	25.02	943.00	963.00	500.09
FROM	NODES	32	TO	33	5.00	1.50	70.59	943.00	950.00	869.33
FROM	NODES	33	TO	32	5.00	1.75	0.00	950.00	935.71	593.88
FROM	NODES	33	TO	34	5.00	1.50	0.00	950.00	920.45	751.07
FROM	NODES	34	TO	33	5.00	1.50	124.40	927.00	950.00	395.45
FROM	NODES	32	TO	34	5.00	1.75	0.00	943.00	920.50	670.00
FROM	NODES	34	TO	32	5.00	1.50	113.92	927.00	943.00	300.16
FROM	NODES	34	TO	35	5.00	1.50	122.47	927.00	913.00	1692.38
FROM	NODES	35	TO	34	5.00	1.50	212.37	913.00	927.00	1969.28
FROM	NODES	38	TO	15	5.00	1.50	203.42	925.00	952.00	1790.25
FROM	NODES	15	TO	36	5.00	1.50	[illegible]	952.00	925.00	1733.55
FROM	NODES	37	TO	36	5.00	1.50	74.03	943.00	925.00	500.05
FROM	NODES	36	TO	37	5.00	1.50	0.00	925.00	911.34	340.56
FROM	NODES	15	TO	39	5.00	1.50	73.99	952.00	895.00	2328.45
FROM	NODES	39	TO	15	5.00	1.50	235.95	895.00	952.00	2211.18
FROM	NODES	39	TO	6	5.00	1.50	233.87	895.00	934.00	1919.67
FROM	NODES	6	TO	39	5.00	1.50	37.28	934.00	895.00	1794.39
FROM	NODES	38	TO	37	5.00	1.50	60.04	913.00	913.00	716.76
FROM	NODES	37	TO	35	5.00	1.75	0.00	913.00	893.50	629.99
FROM	NODES	35	TO	38	5.00	1.50	0.00	933.00	882.40	512.23
FROM	NODES	38	TO	35	5.00	1.50	93.30	887.00	903.00	570.66
FROM	NODES	37	TO	38	5.00	1.50	0.00	913.00	855.27	543.66

FROM NODES 36 TO 37	5.00	1.50	126.69	337.00	915.00	656.22
FROM NODES 37 TO 39	5.00	1.75	0.00	910.00	694.13	775.00
FROM NODES 39 TO 37	5.00	1.50	[illegible]	695.00	910.00	692.57
FROM NODES 38 TO 4	5.00	1.50	[illegible]	587.00	563.95	429.42
FROM NODES 4 TO 38	5.00	1.50	56.56	372.00	667.00	26061.39
FROM NODES 39 TO 4	5.00	1.50	0.00	695.00	672.00	481.79
FROM NODES 4 TO 39	5.00	1.50	105.75	572.00	695.00	200700.18
FROM NODES 39 TO 40	5.00	1.50	0.00	695.00	663.23	207.62
FROM NODES 40 TO 39	5.00	1.50	0.00	684.00	872.28	406.97
FROM NODES 40 TO 4	5.00	1.50	0.00	684.00	670.71	306.33
FROM NODES 4 TO 40	5.00	1.50	0.00	572.00	653.71	200464.00

TOTAL LINK COST= 2483039793.44

DATA FOR THE TRTMT PLANTS IS AS FOLLOWS

PLANT #	FLOW (IMGD)	EFFICIENCY %	COST $1000.
1	126.41	92.00	50971.43
2	126.41	90.00	49134.64
3	126.41	85.00	32949.46
4	126.41	85.00	32949.46

TOTAL PROCESSING COST= 166004991.83

TOTAL SYSTEM COST= 264904785.27

FOR ITERATION # 2 THE FOLLOWING
LINKS COMPRISED THE BRANCHING NETWORK.

	FLOW (CFS)	DIA (FT)	PUMP HP (FT)	U/S INVERT (FT)	D/S INVERT	COST OF LINK $1000.
FROM NODES 6 TO 3	.77	.33	0.00	934.00	883.29	2922.39
FROM NODES 7 TO 3	.73	.83	0.00	943.00	881.53	2293.05
FROM NODES 11 TO 2	.53	.67	0.00	977.00	920.19	1744.04
FROM NODES 9 TO 13	1.10	.83	0.00	965.00	950.25	120.90
FROM NODES 11 TO 12	3.32	1.50	0.00	962.00	954.54	122.00
FROM NODES 13 TO 14	15.37	2.00	0.00	940.00	934.42	400.98
FROM NODES 12 TO 13	7.88	1.75	0.00	952.51	916.63	166.20
FROM NODES 14 TO 2	18.13	1.50	2004.30	970.00	992.00	1955.58
FROM NODES 18 TO 2	6.00	1.00	1450.00	973.00	992.00	1312.11
FROM NODES 10 TO 10	4.73	1.50	0.00	934.91	924.81	913.22
FROM NODES 8 TO 18	1.19	1.00	0.00	953.00	934.91	414.22
FROM NODES 15 TO 16	2.76	1.25	0.00	952.00	940.43	327.46
FROM NODES 17 TO 10	1.53	1.00	0.00	957.00	933.46	684.40
FROM NODES 19 TO 20	8.15	1.75	0.00	968.00	959.53	255.18
FROM NODES 20 TO 2	26.23	1.75	2551.30	973.00	992.00	2416.11
FROM NODES 22 TO 20	14.93	2.50	0.00	962.20	948.03	1912.37
FROM NODES 21 TO 22	13.40	2.00	0.00	980.00	952.22	529.41
FROM NODES 23 TO 24	1.19	1.00	0.00	1002.00	961.97	522.72
FROM NODES 24 TO 1	4.49	.67	1942.85	1017.00	1012.00	1604.43
FROM NODES 25 TO 1	11.30	2.25	0.00	1008.00	990.87	2309.79
FROM NODES 26 TO 1	7.94	1.75	0.00	1031.00	1014.76	601.94
FROM NODES 31 TO 21	7.49	1.00	1801.50	[illegible]	981.00	1394.08
FROM NODES 28 TO 25	3.42	.67	792.84	987.00	1008.00	379.05
FROM NODES 30 TO 31	1.37	1.00	0.00	960.00	939.09	515.46
FROM NODES 33 TO 31	2.76	1.25	0.00	950.00	933.00	601.15
FROM NODES 29 TO 28	1.13	1.00	0.00	963.00	922.34	1881.05
FROM NODES 27 TO 28	.79	.83	0.00	985.00	935.69	1339.70
FROM NODES 5 TO 28	.55	.67	0.00	977.00	913.77	1942.48
FROM NODES 32 TO 34	.40	.83	0.00	943.00	857.71	1525.63
FROM NODES 34 TO 32	.74	.83	0.00	867.71	733.69	23117.39
FROM NODES 35 TO 37	3.94	1.50	0.00	925.00	912.84	545.54
FROM NODES 36 TO 36	3.55	.67	2054.45	903.00	887.00	396.71
FROM NODES 37 TO 38	6.31	1.75	0.00	920.64	887.00	730.11
FROM NODES 38 TO 4	11.80	2.00	0.00	887.00	888.06	592.40
FROM NODES 39 TO 4	1.58	1.00	0.00	895.00	883.51	404.08
FROM NODES 40 TO 4	41.37	1.00	0.00	884.00	880.01	285.20

TOTAL LINK COST= 5983-765.97

DATA FOR THE TRTMT PLANTS IS AS FOLLOWS

PLANT #	FLOW (IMGD)	EFFICIENCY %	COST $1000.
1	31.63	87.00	1553.06
2	72.28	87.00	29815.54
3	3.09	65.00	1289.33
4	20.40	87.00	10978.67

TOTAL PROCESSING COST= 5759-399.77

TOTAL SYSTEM COST= 117428865.74

FOR ITERATION # 3 THE FOLLOWING
LINKS COMPRISED THE BRANCHING NETWORK.

				FLOW (CFS)	DIA (FT)	PUMP HP (FT.)	U/S INVERT (FT)	D/S INVERT	COST OF LINK $1000.
FROM NODES	6	TO	7	.77	.83	0.00	934.00	904.75	567.14
FROM NODES	7	TO	5	1.55	2.00	0.00	934.75	856.10	4850.02
FROM NODES	10	TO	9	.59	.67	0.00	977.00	945.50	576.16
FROM NODES	8	TO	9	1.16	1.00	0.00	955.00	922.52	516.11
FROM NODES	9	TO	13	2.96	.67	211.92	965.00	970.00	267.06
FROM NODES	11	TO	12	5.52	1.50	0.00	960.00	954.51	122.16
FROM NODES	13	TO	14	17.15	2.25	0.00	940.63	938.39	499.44
FROM NODES	12	TO	13	7.88	1.75	0.00	954.51	940.01	326.00
FROM NODES	14	TO	2	19.91	1.50	2490.92	978.00	992.00	2206.15
FROM NODES	15	TO	2	5.92	1.75	0.00	973.00	957.97	363.15
FROM NODES	16	TO	18	4.25	.67	564.24	903.00	973.00	279.19
FROM NODES	15	TO	16	2.76	1.25	0.00	952.00	940.48	567.46
FROM NODES	17	TO	19	1.33	1.00	0.00	957.00	927.11	490.00
FROM NODES	19	TO	20	9.70	2.00	0.00	937.11	931.22	595.00
FROM NODES	20	TO	2	31.87	1.75	4399.10	975.00	992.00	3513.18
FROM NODES	22	TO	20	18.99	1.50	1800.02	990.00	970.00	1849.54
FROM NODES	21	TO	22	15.93	2.25	0.00	983.00	966.58	563.92
FROM NODES	23	TO	22	1.49	1.00	0.00	1002.00	979.94	343.53
FROM NODES	24	TO	1	3.00	.67	702.73	1017.00	1042.00	356.55
FROM NODES	23	TO	1	17.20	2.25	0.00	1031.00	1010.91	797.58
FROM NODES	25	TO	26	9.32	1.00	1577.11	1008.00	1031.00	1319.78
FROM NODES	31	TO	21	10.02	1.00	2134.54	970.00	983.00	1769.22
FROM NODES	26	TO	25	1.30	1.00	0.00	936.69	961.08	4356.94
FROM NODES	30	TO	31	4.10	.67	1138.20	960.00	970.00	943.99
FROM NODES	33	TO	31	2.70	1.25	0.00	950.00	960.00	631.15
FROM NODES	29	TO	30	2.53	1.25	0.00	820.28	802.98	6537.08
FROM NODES	27	TO	28	.79	.83	0.00	985.00	958.69	1339.70
FROM NODES	5	TO	29	.55	.67	0.00	977.00	919.04	1112.35
FROM NODES	32	TO	29	.79	.83	0.00	851.71	820.28	5392.74
FROM NODES	34	TO	32	.40	.83	0.00	927.00	851.71	2567.35
FROM NODES	36	TO	37	3.94	1.50	0.00	925.00	913.84	343.34
FROM NODES	35	TO	38	2.76	1.25	0.00	903.00	879.00	484.22
FROM NODES	37	TO	38	5.31	1.75	0.00	910.54	887.00	738.21
FROM NODES	38	TO	4	10.81	2.00	0.00	879.00	861.48	713.39
FROM NODES	39	TO	40	1.33	1.00	0.00	895.00	877.75	252.83
FROM NODES	40	TO	4	3.55	1.50	0.00	877.76	864.36	390.36

TOTAL LINK COST= 49135929.84

DATA FOR THE TRTMT PLANTS IS AS FOLLOWS

PLANT #	FLOW (IMGD)	EFFICIENCY %	COST $1000.
1	27.29	84.00	13576.52
2	77.69	87.00	31565.67
3	2.09	35.00	1269.35
4	19.34	90.00	11140.91

TOTAL PROCESSING COST= 57572627.80

TOTAL SYSTEM COST= 106708557.64

FOR ITERATION # - THE FOLLOWING
LINKS COMPRISED THE BRANCHING NETWORK.

	FLOW (CFS)	DIA (FT)	PUMP HP (FT)	U/S INVERT (FT)	D/S INVERT	COST OF LINK $1000.
FROM NODES 5 TO 3	.77	.83	0.00	934.00	868.29	2923.39
FROM NODES 7 TO 3	.78	.83	0.00	943.00	881.53	2298.05
FROM NODES 10 TO 9	.59	.67	0.00	977.00	945.53	378.13
FROM NODES 9 TO 13	1.75	1.00	0.00	945.30	908.25	378.08
FROM NODES 11 TO 12	5.52	1.50	0.00	960.00	954.51	122.06
FROM NODES 13 TO 14	15.97	2.25	0.00	933.25	901.11	597.04
FROM NODES 12 TO 13	7.80	1.75	0.00	954.51	940.88	326.22
FROM NODES 14 TO 1	15.73	1.50	2179.37	978.00	992.88	2266.17
FROM NODES 15 TO 2	7.20	.83	1390.45	973.00	992.81	1509.92
FROM NODES 16 TO 18	4.73	1.50	0.00	934.91	924.81	913.22
FROM NODES 8 TO 10	1.18	1.00	0.00	955.01	934.91	410.22
FROM NODES 15 TO 18	2.70	1.25	0.00	952.00	940.43	327.46
FROM NODES 17 TO 19	1.50	1.00	0.00	957.00	937.11	491.01
FROM NODES 19 TO 20	9.73	2.00	0.00	937.11	931.22	655.00
FROM NODES 20 TO 2	31.97	1.75	4399.10	973.00	992.00	3518.05
FROM NODES 22 TO 20	18.99	1.50	2011.32	990.00	978.00	1926.98
FROM NODES 21 TO 22	15.93	2.25	0.00	952.44	939.02	1126.22
FROM NODES 23 TO 22	1.49	1.00	0.00	1002.00	979.94	340.53
FROM NODES 24 TO 1	3.00	.67	702.70	1047.00	1042.03	356.55
FROM NODES 26 TO 1	17.20	2.25	0.00	1031.00	1013.90	797.58
FROM NODES 25 TO 26	9.31	1.00	1577.12	1008.00	1031.00	1319.78
FROM NODES 31 TO 21	10.02	2.00	0.00	970.00	952.44	900.11
FROM NODES 28 TO 25	1.05	1.00	0.00	936.69	921.88	4156.94
FROM NODES 30 TO 31	3.31	.67	630.33	960.00	970.00	649.42
FROM NODES 33 TO 31	3.55	.67	824.22	950.00	970.00	765.32
FROM NODES 29 TO 30	1.74	1.00	0.00	919.04	898.50	1043.16
FROM NODES 27 TO 28	.79	.83	0.00	985.01	936.69	1339.78
FROM NODES 5 TO 29	.55	.67	0.00	977.00	919.04	1122.35
FROM NODES 32 TO 33	.40	.83	0.00	943.00	882.11	1466.94
FROM NODES 34 TO 33	.40	.83	0.00	927.00	841.23	3336.76
FROM NODES 36 TO 37	3.94	1.50	0.00	925.00	910.84	343.04
FROM NODES 35 TO 38	2.70	1.25	0.00	903.00	879.60	484.22
FROM NODES 37 TO 38	6.31	1.75	0.00	910.84	887.00	733.21
FROM NODES 38 TO 4	10.81	2.00	0.00	879.60	861.42	710.39
FROM NODES 39 TO 40	1.28	1.40	0.00	895.00	877.75	252.88
FROM NODES 40 TO 4	3.55	1.50	0.00	877.75	864.30	390.36

TOTAL LINK COST= 41340798.51

DATA FOR THE TRTMT PLANTS IS AS FOLLOWS

PLANT #	FLOW (MGD)	EFFICIENCY %	COST $1000.
1	27.29	84.00	13576.52
2	77.69	87.00	51565.87
3	2.69	85.00	1269.33
4	19.34	90.00	11140.91

TOTAL PROCESSING COST= 57572627.80

TOTAL SYSTEM COST= 98913426.31

FOR ITERATION # 5 THE FOLLOWING LINKS COMPRISED THE BRANCHING NETWORK.

				FLOW (CFS)	DIA (FT)	PUMP HP (FT)	U/S INVERT (FT)	D/S INVERT	COST OF LINK $1000.
FROM NODES	6	TO	7	.77	.83	0.00	954.00	954.75	537.04
FROM NODES	7	TO	8	1.55	1.00	0.00	954.75	850.70	358.82
FROM NODES	10	TO	9	.59	.67	0.00	977.00	945.50	376.10
FROM NODES	8	TO	9	1.18	1.00	0.00	955.00	952.22	360.21
FROM NODES	9	TO	13	2.95	.67	211.92	965.00	970.00	187.06
FROM NODES	11	TO	12	5.52	1.50	0.00	960.00	954.52	122.06
FROM NODES	10	TO	11	9.47	1.75	0.00	973.00	960.81	211.79
FROM NODES	12	TO	14	7.86	1.75	0.00	954.51	942.19	820.05
FROM NODES	14	TO	2	19.91	1.50	2477.81	975.00	992.00	2254.67
FROM NODES	15	TO	2	5.91	1.75	0.00	973.00	987.97	386.15
FROM NODES	16	TO	18	3.35	.67	584.24	963.00	973.00	379.09
FROM NODES	15	TO	16	2.70	1.25	0.00	952.00	940.43	327.46
FROM NODES	17	TO	19	.55	1.00	0.00	957.00	937.11	490.03
FROM NODES	19	TO	20	9.70	2.00	0.00	937.11	931.22	833.00
FROM NODES	20	TO	22	32.95	2.00	2667.57	973.00	992.00	2053.30
FROM NODES	21	TO	22	19.76	1.50	2245.97	990.00	978.00	1174.70
FROM NODES	22	TO	21	10.72	2.25	0.00	949.57	964.73	1216.61
FROM NODES	23	TO	22	1.49	1.00	0.00	1002.00	979.94	348.55
FROM NODES	24	TO	1	3.00	.67	762.73	1017.00	1042.[illegible]	356.55
FROM NODES	26	TO	1	10.47	2.25	0.00	1031.00	1012.73	770.40
FROM NODES	25	TO	26	8.53	1.00	1256.30	1003.00	1031.00	1105.56
FROM NODES	31	TO	25	10.99	2.00	0.00	973.00	949.57	971.78
FROM NODES	28	TO	25	.59	.67	0.00	987.00	985.50	1505.27
FROM NODES	30	TO	31	4.10	.67	1855.57	965.00	970.00	384.34
FROM NODES	33	TO	30	2.55	.67	824.22	955.00	970.00	766.22
FROM NODES	29	TO	33	2.53	1.25	0.00	915.68	898.35	1703.75
FROM NODES	27	TO	29	.79	.83	0.00	985.00	915.05	1820.35
FROM NODES	5	TO	29	.55	.67	0.00	977.00	919.04	1122.05
FROM NODES	32	TO	33	.40	.83	0.00	943.00	882.11	1106.94
FROM NODES	34	TO	33	.40	.83	0.00	927.00	841.23	3356.76
FROM NODES	36	TO	37	3.94	1.50	0.00	925.00	900.84	346.04
FROM NODES	35	TO	38	2.70	1.25	0.00	903.00	879.00	484.22
FROM NODES	37	TO	38	6.31	1.75	0.00	910.04	887.00	733.24
FROM NODES	38	TO	4	16.81	2.00	0.00	879.60	861.42	723.09
FROM NODES	39	TO	40	1.53	1.00	0.00	895.00	867.76	252.03
FROM NODES	40	TO	4	3.55	1.50	0.00	877.70	864.55	391.30

TOTAL LINK COST= 3830787.07

DATA FOR THE TRTMT PLANTS IS AS FOLLOWS

PLANT #	FLOW (IMGD)	EFFICIENCY %	COST $1000.
1	26.20	87.00	13380.29
2	78.75	87.00	31900.96
3	2.09	35.00	1269.53
4	19.34	84.00	10341.62

TOTAL PROCESSING COST= 5691895.54

TOTAL SYSTEM COST= 9522682.61

APPENDIX I

SUMMARY OF NOTATION

The following is a complete list of all variables used in the program. A brief description of each variable is supplied.

A( ) = matrix of structural coefficients (possible values 1, 0 or -1).

ACRES( ) = array containing values of acreage for each node.

AREA( ) = cross sectional area of pipe.

BMAX( ) = array holding values of maximum flowrates at end of design period for each node.

BMIN( ) = array holding values of minimum flowrates at beginning of design period for each node.

BODTOP( ) = value of BOD concentration at the top of a reach.

BPLANT( ) = array containing predetermined plant capacities.

C( ) = array containing complete set of cost coefficients.

COMBEG( ) = array holding values of commercial flowrates at beginning of design period for each node.

COMEND( ) = array holding values of commercial flowrates at end of design period for each node.

COST( ) = array containing total link costs for various diameters for a specific link.

CPEN( ) = array holding values of penalty cost coefficients.

CPROD( ) = array containing plant costs for use in calculating final cost.

DALLOW = size of smallest diameter allowable due to possible upstream constraints.

DALLOW = maximum allowable oxygen deficit.

DC = largest oxygen deficit along any portion of a reach.

DDISC( ) = array holding sizes of selected pipe diameters.

DECOPT( ) = matrix containing optimum treatment efficiency for each level of oxygen deficit for each plant.

DEFICT( ) = array containing discrete levels of oxygen deficit at the end of the reach.

DEFTOP = value of the oxygen deficit at the top of a reach.

DISCE = discrete value of the oxygen deficit at the end of a reach.

DMIN = minimum allowable pipe diameter.

DOPT( ) = array holding optimum diameters for each link for a specific iteration.

DRATIO = ratio of partial full depth to full flow depth.

DROP = drop manhole height.

DS = increment for levels of oxygen deficit.

DSOL( ) = array containing treatment efficiencies of each plant which correspond to the minimum cost for each iteration.

E = calculated value of the oxygen deficit at the end of a reach.

EDS( ) = elevation at top of pipe at downstream end of reach.

EFF( ) = array containing allowable treatment plant efficiencies.

EUS( ) = elevation at top of pipe at upstream end of reach.

F = value of BOD concentration at the end of a reach.

FD( ) = array containing cost for particular combination of water quality and treatment efficiency.

FLOW = initial value of flowrate for each link.

FMAX( ) = array containing current lowest costs for each level of oxygen deficit for a specific reach.

FO = initial value of BOD concentration just upstream of first plant.

FOPT( ) = matrix containing individual treatment plant costs which correspond to the minimum cost.

FSOL( ) = array containing costs of each plant which correspond to the minimum cost for each iteration.

FUS( ) = matrix containing BOD concentrations at the end of the reach.

HP( ) = array containing values of installed horsepower for each link.

IEFF( ) = array holding values of optimum treatment efficiencies for each level of oxygen deficit.

IFLAG( ) = array used to indicate whether or not the optimum diameters for all links upstream of a specific link have been calculated.

ITER = iteration number.

L = number of transportation links.

LZ = total number of flow variables, excluding slack variables.

N = number of nodes in the network.

NAMES( ) = array containing identification names for each node.

NCEN = number of treatment centres.

ND = number of possible treatment efficiencies.

NDS( ) = array holding downstream node numbers for each link.

NPIPES = number of allowable pipe diameters.

NQ = total number of flow plus slack variables.

NSTAGE = number of reaches.

NSTATE = number of possible levels of discrete oxygen deficits.

NUS( ) = array holding upstream node numbers for each link.

OPTF( ) = matrix containing the BOD concentrations which correspond to the optimum plant efficiencies for each reach.

POPBEG( ) = array holding population at the beginning of the design period for each node.

POPEND( ) = array holding population at the end of the design period for each node.

QAVG = supply flowrate per person.

QF = flowrate for pipes flowing full at minimum allowable velocity.

QFULL = flowrate for pipe flowing full at maximum allowable velocity.

QMAX( ) = array containing design flows for links and slack variable for treatment plants.

QPLANT( ) = array containing design flows for links and slack variable for treatment plants.

QPLANT = value of flowrate at a treatment plant.

QRATIO = maximum flowrate divided by flowrate for pipe flowing full.

QRIV = value of river flow upstream of treatment facilities.

QS = minimum pipe flowrates at beginning of design period (subroutine VMINN) or maximum pipe flowrate at end of design period (subroutine VMAXX).

R( ) = array holding values of reoxygenation rates for each reach.

RINDUS( ) = array containing values of industrial flowrates at the beginning of the design period for each node.

RINEND( ) = array containing values of industrial flowrates at the end of the design period for each node.

RINFLT = value of infiltration per acre.

RTN( ) = array holding cost of treatment plant for various efficiencies and a known flowrate.

S = calculated slope for givin diameter and flow (partial full)

SAVAIL = slope which results in minimum cover at downstream end of link.

SC = slope of sewer carrying QS and a self cleansing as a sewer flowing full with velocity VMIN.

SF = required slope to attain VMIN.

SIGN = variable set equal to -1.0.

SIZPIP = inside pipe diameter.

SMNLMT = slope which is required to satisfy minimum velocity constraint.

STATE( ) = array holding ground elevations for each node.

S2 = maximum slope which just satisfies maximum velocity constraint.

T( ) = array containing oxygen deficit for treated waste at each plant.

T = critical shear stress.

TREACH( ) = array holding travel time for each reach.

TYPE = variable set equal to -1.0.

USDIA = value of the largest upstream diameter for a specific link.

VFULL = the full flow velocity.

VMAX = maximum allowable velocity.

VMAXP = velocity in sewer (partially full) for maximum flowrate.

VMIN = minimum full flow velocity based on the critical shear stress approach.

VRATIO = ratio of partial full velocity to full flow velocity.

W = specific weight of wastewater.

X( ) = array holding update flowrate values.

XCOST = minimum cost for a link (force main).

XINFNT( ) = array holding incoming BOD concentrations to each plant.

XK( ) = array holding values of deoxygenation rates for each reach.

XL( ) = array holding transportation distance for each link.

XMNCOV = allowable minimum cover.

XMNCST = minimum cost for a link (gravity).