Publication No. 104

Hueristic Method for the Design of Stormwater Drainage Systems Using LOTUS 1-2-3

by

Stewart Wayne Miles

University of Florida
Gainesville, Florida 32611

Publication No. 104

HEURISTIC METHOD FOR THE DESIGN OF STORMWATER DRAINAGE SYSTEMS USING LOTUS 1-2-3

BY

STEWART WAYNE MILES

A THESIS PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN
PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF ENGINEERING

UNIVERSITY OF FLORIDA

1988

Publication No. 104

HEURISTIC METHOD FOR THE DESIGN OF STORMWATER DRAINAGE SYSTEMS USING LOTUS 1-2-3

BY

STEWART WAYNE MILES

A THESIS PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN
PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF ENGINEERING

UNIVERSITY OF FLORIDA

1988

TABLE OF CONTENTS

APPENDIX

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to express my sincere appreciation to everyone in the Department of Environmental Engineering who has helped in making the completion of this thesis possible. I would also like to extend my greatest thanks to my parents who have given me constant love, encouragement, and support throughout my education.

My committee chairman, Dr. James Heaney, has provided boundless guidance, support, and friendship through my graduate studies and for this I am genuinely thankful. I would also like to express my thanks to the other members of my committee, Dr. Wayne Huber and Dr. Warren Viessman, who provided valuable advice and direction during my studies at the University of Florida.

The funding for the research which led to this thesis was provided by the U.S. Geological Survey Annual Allotment to the Florida Water Resources Research Center. The assistance of the Florida Department of Transportation is also appreciated.

The many friends I have made during my years in this department are also deserving of thanks for their contributions. Many of the ideas in this thesis are variations or incorporations of thoughts and suggestions from students and co-workers in the department.

Lastly, I want to express my gratitude to my fiancee, Jessica, for her ceaseless patience and love and for helping me to build self-confidence and a desire to progress.

TABLE OF CONTENTS

page

CHAPTERS

APPENDIX

LIST OF TABLES

LIST OF FIGURES

Abstract of Thesis Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirement for the Degree of Master of Engineering

HEURISTIC METHOD FOR THE DESIGN OF STORMWATER DRAINAGE SYSTEMS USING LOTUS 1-2-3

By

S. Wayne Miles

April, 1987

Chairman: Dr. James P. Heaney
Major Department: Environmental Engineering Sciences

The Lotus 1-2-3 spreadsheet package has been used to develop a stormwater drainage design method which is based on the Florida Department of Transportation (FDOT) hand tabulation form. By using the FDOT procedure, the spreadsheet method allows engineers already familiar with this calculation method to more easily adapt to the new environment. The spreadsheet performs the necessary calculations quickly and easily while not hindering the user with tedious input procedures and confusing algorithms typical of many computer codes. The spreadsheet method also employs a heuristic cost estimation approach which finds the least cost design.

The heuristic approach is based on a cost estimate derived using the FDOT itemized average bid database for highway construction items. Piping, structural, and excavation costs are included in the cost estimate. The Thomasville Highway project has been redesigned as a case study with a cost estimate of the resulting design of 10% less than that of the original design.

This spreadsheet design procedure has been used to solve a problem previously solved with two dynamic programming algorithms. Since the spreadsheet method does not use a sophisticated optimization algorithm, many simplifying assumptions made in the dynamic programming algorithms are not necessary. The results show that the spreadsheet method was able to significantly improve on the dynamic programming designs while more consistently meeting the design criteria.

The spreadsheet design template has also been linked with a spreadsheet based preprocessor for the EPA Stormwater Management Model (SWMM) Runoff Module. The Runoff Module is used to predict peak flow values for the system design and is used as verification of the final drainage design. The results of an example problem using this procedure are presented.

CHAPTER 1
INTRODUCTION

Research in the field of stormwater drainage design has brought steady improvements in design procedures through the use of computerized simulation and optimization models. The advancements in the power of microcomputers in recent years have brought extensive computational capabilities to many engineers. The use of these new tools and procedures for actual designs, however, often lags far behind. Because of precedent and constraints in time and money, many engineers continue to use well-known and established manual design procedures.

This thesis proposes alternative methods for the design of stormwater drainage systems that take advantage of modern tools and simulation techniques presently available while not using complex mathematical algorithms. The design procedure described in this thesis is based on the current Florida Department of Transportation (FDOT) drainage design procedure as described in the Drainage Manual (FDOT, 1987). The procedure is performed on the Lotus 1-2-3 (Lotus Development Corporation, 1986) spreadsheet program and replicates the FDOT hand tabulation form. The spreadsheet procedure permits engineers to take advantage of the computational power of the microcomputer while working with a familiar design format.

When design calculations are performed manually, the number of alternatives evaluated is severely limited. The most economical design is often not obtained because it is only practical to investigate a few alternatives (Desher and Davis, 1986). The intent of this thesis is to present a methodology whereby design procedures are improved and computerized in a manner that smoothly takes engineers from their current manual methods to more refined approaches. This process will necessarily involve incremental changes in the current design procedures. The spreadsheet provides a computerized environment where these incremental changes may take place.

The spreadsheet design method has all of the traditional advantages of a computerized design technique such as speed of computation, but lacks many of the drawbacks associated with computer algorithms such as tedious problem input procedures, difficult constraint definition, and difficulty of understanding by others. Commonly, computerized methods for the drainage design problem employ mathematical optimization methods (e.g. linear programming, non-linear programming, dynamic programming). These methods are applied to simplified and well-behaved versions of the true design problem. It is difficult to define the system constraints of a real design problem to fit into the structure of an optimization problem. In the description of their dynamic programming algorithm, Merritt and Bogan (1973, p. 36) admit that "It is unlikely that any optimization method achieves a true optimum when the full scope of a real world setting is considered." The spreadsheet design method does not employ a rigorous optimization technique. Rather it is simply an environment where all traditional hydraulic calculations may be easily tabulated. This feature allows the spreadsheet method to retain the flexibility of giving a feasible solution to any design problem. The spreadsheet also offers the ability to perform many trials in a stormwater system design which encourages the engineer to fine-tune the pipe sizes and slopes in a drainage system to find the least-cost design.

Chapter 2 of this thesis describes the mechanics involved in performing the FDOT Tabulation Method on the spreadsheet. In Chapter 3 the spreadsheet method is compared to two well-known dynamic programming algorithms and the results of an example problem that is solved with both the dyamic programming algorithms and the spreadsheet method are presented. Chapter 4 presents the design template capability of performing itemized cost estimates of a system design during the design procedure. A description of the powerful capability of accessing an external database from within the spreadsheet is also described in this chapter. Chapter 5 describes a preprocessor developed for the EPA Stormwater Management Model (SWMM)(Huber et al., 1981; Roesner et al., 1983). The preprocessor has been linked to the drainage design template in order to use SWMM as an alternative method to determine design flows in the pipe network and as a method of verification of the final drainage design.

CHAPTER 2
DESCRIPTION OF SPREADSHEET TABULATION FORM

Background of Study

This study began with the intention of designing an experiment to determine whether the use of computerized models in designing stormwater drainage systems could be economically justified. In order to conduct such an experiment, past projects must be analyzed and redesigned using new procedures. The search for past projects led to the Florida DOT.

The FDOT has drainage design procedures which are very well documented in their Drainage Manual (FDOT, 1987). They also have a large number of past project designs available in blueprint form as well as planning calculation form. The most extensive database of their past projects, however, is found in their cost estimating department. An itemized unit cost which is based on average bid prices from past projects is available for each highway construction item. This database is updated every six months and the item numbering system allows drainage related items to be determined easily.

The FDOT Drainage Manual (1987) lists a mainframe Fortran program called "Draino" (PEGDRG32) as available to assist in drainage design. This program uses a heuristic algorithm that minimizes pipe costs of a drainage system. The first goal of the project was to modify the program so it could be run on a personal computer and to replicate the program algorithm on the spreadsheet. The former was performed by Potter (1986) and the latter by Miles (1986). Once this material was presented to the FDOT for feedback, it was learned that the "Draino" program was seldom used. FDOT personnel found the mainframe based program difficult to apply to problems with real world constraints and found that the input procedure was very tedious. Since the program assumes that all pipes are flowing full in the system, users found that it had a limited number of projects to which it was applicable. The program did show, however, that the FDOT was interested in optimizing their drainage system designs. This realization led to a new design approach on the spreadsheet.

Literature Review

The history of computerized design algorithms for sewer systems dates back over two decades to papers by Liebman (1967) and Holland (1966). These early algorithms were primarily based on mathematical optimization techniques. These algorithms made very simplifying assumptions about the problem definition and the attention was focused on the solution technique. Liebman's linear programming algorithm only dealt with the network layout problem, and Holland's nonlinear algorithm could not handle discrete pipe sizes.

By the early seventies the most common technique applied to the sewer design problem was dynamic programming. Meredith (1972), Merritt and Bogan (1973), Tang et al. (1975), and Mays and Wenzel (1976) resented variations of the dynamic programming algorithm to analyze the sewer design problem. These algorithms, however, continued to make simplifying assumptions concerning the problem definition and criteria. All of the mentioned algorithms used cost functions as the optimization criteria and all required powerful mainframe computers with extensive storage capabilities (Desher and Davis, 1986).

Recently, models have been developed to perform sewer design on the microcomputer (Desher and Davis, 1986). These models have taken a heuristic design approach, but still retain a batch run format which keeps all computations hidden from the user.

A 1984 survey showed that 86% of water resources professionals from state agencies and private consultants had used mathematical models for the planning, design, and operation of water resources systems (Austin, 1986). Austin also reveals, however, that simulation models are used more often than optimization models. Simulation models are helpful, but are awkward for design because of their batch run format. However, a reliable simulation model used for verification of a simple optimization technique would be very valuable for system design.

The latest trend has been to focus research on the development and transfer of tools and skills rather than providing solutions to problems. Fedra and Loucks (1985) propose that the recent advancements in the power of computers be applied to the problem of interaction between people and machines rather than on more advanced problem solution techniques. They present the basic premise as the organization of information to facilitate the use of human judgement. The spreadsheet is a tool which follows this basic premise. The spreadsheet is usually not the most efficient method of problem solving available; however, it is a very good method of performing calculations and simple solution algorithms in an orgainized manner.

Problem Description

The initial task in developing a spreadsheet design method was to address the concern of finding a least-cost design for the vertical alignment of the stormwater drainage system. A profile view of this design problem is shown in Figure 2-1. The design parameters for this problem are the pipe sizes, pipe slopes, and manhole drops through the system. The peak design flows in the system are calculated by first estimating the inflow into each inlet based on subcatchment characteristics and a design rainfall event. The next step is to route the inflow hydrographs (or peak flows, depending on the sophistication of the method) through the system in order to determine the maximum required capacity of each pipe. These steps may be performed with the use of a simple empirical model such as the Rational Method, or may be performed by a complex hydrologic model. The manhole locations and ground elevations at each location are assumed given. The system must be designed such that the pipes are able to carry the peak design flow capacity while also meeting all other constraints. Also, maximum and minimum velocities in the pipe are prescribed. The velocity must be above a minimum to prevent clogging and below a maximum to decrease scouring which may reduce the lifetime of the pipe. Lastly, the hydraulic grade line must be addressed. The FDOT procedure allows that only the frictional losses in the pipe be calculated as long as the hydraulic grade line does not exceed a height of one foot below the ground elevation at a manhole junction. For structural reasons, it is also necessary that a minimum cover be kept between the pipe crown and the ground elevation.

The trade-off in this problem is between pipe size and excavation. To fulfill the desired flow capacity in a pipe, a larger pipe may be used at a shallower slope thereby saving excavation cost but having a larger pipe cost. The other alternative is to use a smaller and less-

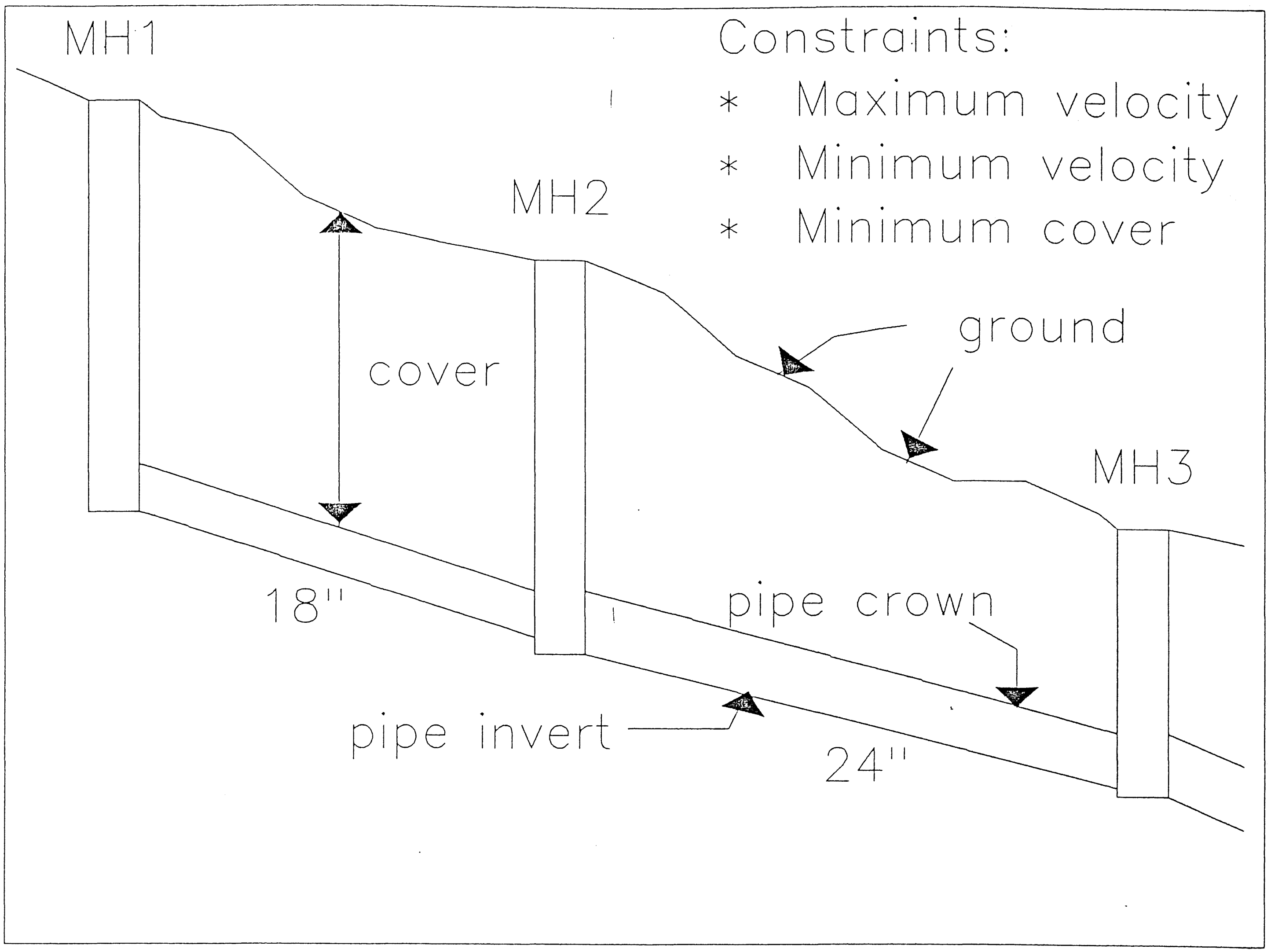


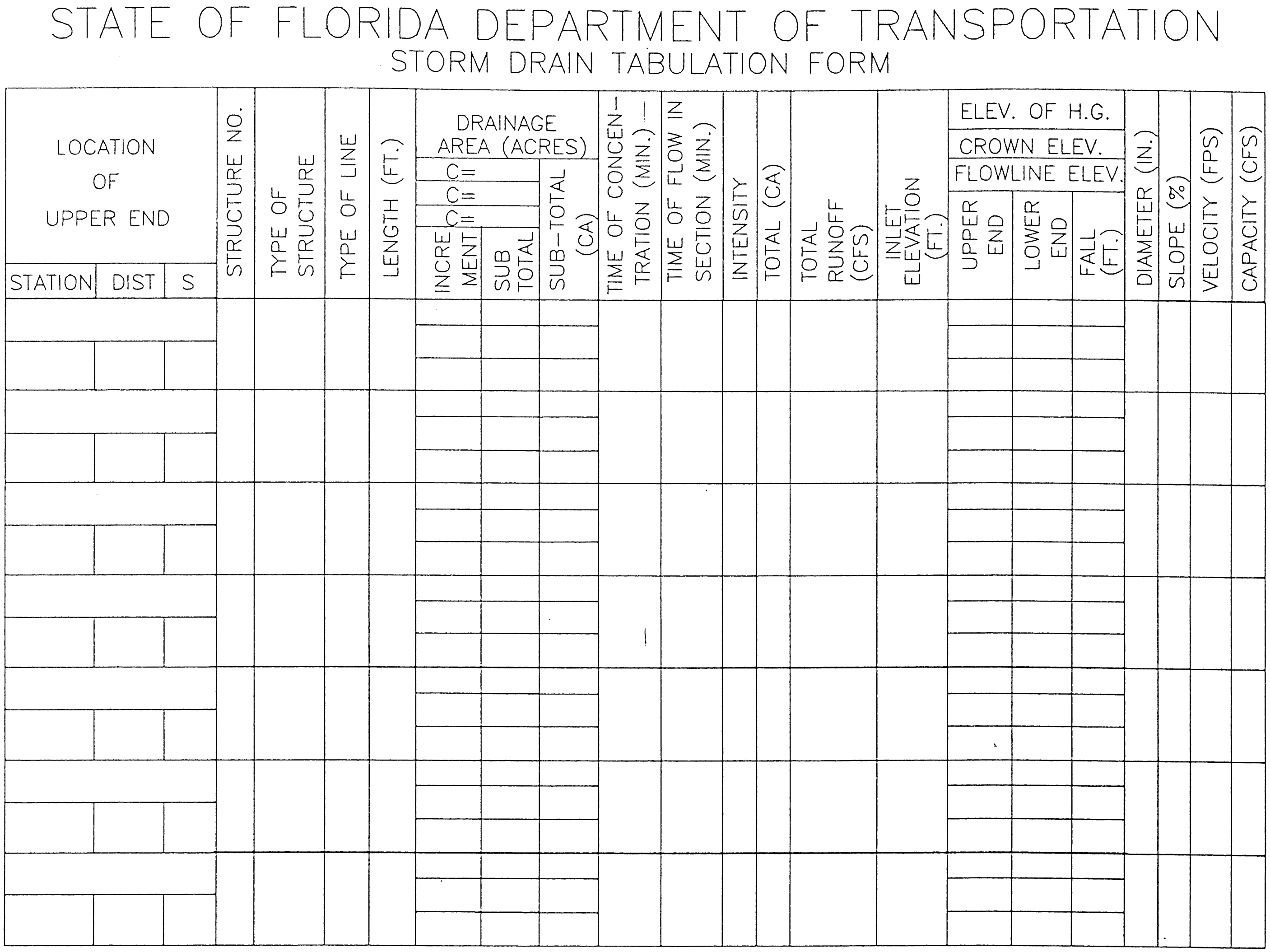
Figure 2-1. Profile View of Vertical Alignment of Stormwater Drainage System

costly pipe at a steeper slope but with higher excavation costs. Analysis of the trade-off has been performed using many different problem solving techniques. The majority of these techniques have been based on mathematical optimization algorithms. An overview of this research and specifically the technique of dynamic programming is given in Chapter 3.

Current Methods of Design

Design methods used by drainage engineers have changed very little in the last 50 years (Desher and Davis, 1986). It is still common practice to use nomographs and task specific slide calculators for determining pipe sizes, pipe slopes, and velocities in a design.

Presently, the FDOT personnel perform most of their drainage calculations using the worksheet shown in Figure 2-2. Explanations of a few of the entries needed on this tabulation form are given in an excerpt from the FDOT Drainage Manual (1987) shown in Table 2-1. These calculations are most commonly performed by hand or with a nomograph. The Rational Method is used to determine the runoff volumes from each drainage area. The rainfall intensity is determined by the use of intensity-duration-frequency curves given the design frequency and assuming the storm duration equal to the time of concentration for the pipe. This effectively produces an empirical flow routing scheme since the time of concentration tends to increase for pipes in the downstream direction which increases the intensity of the storm used for the flow calculation in that pipe. This result increases the difficulty of performing this calculation by hand since a change in any upstream pipe can potentially change the design flow in all downstream pipes. The intensity increases in this case because of the assumption that the time of concentration is



STATE OF FLORIDA DEPARTMENT OF TRANSPORTATION
STORM DRAIN TABULATION FORM

LOCATION OF UPPER END			STRUCTURE NO.	TYPE OF STRUCTURE	TYPE OF LINE	LENGTH (FT.)	DRAINAGE AREA (ACRES)			TIME OF CONCEN-TRATION (MIN.)	TIME OF FLOW IN SECTION (MIN.)	INTENSITY	TOTAL (CA)	TOTAL RUNOFF (CFS)	INLET ELEVATION (FT.)	ELEV. OF H.G. CROWN ELEV. FLOWLINE ELEV.			DIAMETER (IN.)	SLOPE (%)	VELOCITY (FPS)	CAPACITY (CFS)
STATION	DIST	S					C= C= C= INCRE MENT	SUB TOTAL	SUB-TOTAL (CA)							UPPER END	LOWER END	FALL (FT.)				

Figure 2-2. FDOT Recommended Drainage Design Tabulation Form (1987)

Table 2-1. Excerpt From Description of Tabulation Form From Drainage Manual (FDOT, 1987, p. 10-23).

---

14. Time of Flow in Section (min)

    This is the time it takes the runoff to pass through the section of pipe in question; it depends on the velocity as well as the condition of flow (i.e., gradient or physical flow time based on proper condition and velocity).

15. Intensity

    Intensity values are determined from one of the 11 intensity-duration-frequency (IDF) curves developed by the Department and presented in Chapter 5 of this volume. Intensity depends on the design frequency and the time of concentration.

16. Total (CA)

    The total CA is the sum of the subtotal CAs.

17. Total Runoff (cfs)
    Total runoff is the product of the intensity and the total CA, less inlet bypass and exfiltration.

18. Inlet Elevation (feet)

    This column lists the elevation of the gutter if the structure is a curb inlet. In the case of manholes and ditch bottom inlets, either the top or grate elevation and the slot elevation are shown and so noted.

related to the velocity of the water in the pipe. This is a common mistake made when using the Rational Method (Cunningham, 1987). [The Kinematic Wave equation, a method of determining the time of concentration based upon the travel time of a wave, is presented in the Drainage Manual as a method to determine overland flow time of concentration, but is not mentioned with respect to determining the time of concentration in a pipe segment (FDOT, 1987).] A change in an upstream pipe size will change the velocity in that pipe which may potentially change the time of concentration of the downstream pipe. This change is stated as potentially changing the time of concentration since, in an effort to employ a conservative design, the maximum time of concentration of all inflowing streams is used as the time of concentration for each segment. The change in the time of concentration affects the duration of the storm used to determine the rainfall intensity from the intensity-duration-frequency curves, which in turn, changes the design flow calculated by the Rational Method. In this manner, one change in an upstream pipe size or slope results in a re-evaluation of the calculations for the entire downstream portion of the pipe system.

Repetitive hand calculations are subject to error and do not encourage the engineer to "push the limits" of the design criteria to find an optimal design. The tabulation form procedure also does not explicitly include system cost as a design criteria. Therefore, in addition to having no incentive, the engineer has no indicator to determine that a design may be "improved."

The tabulation form has become the accepted practice for highway drainage design. Designs obtained with the use of a computer model or alternative method must be compared to the tabulation form design. With the time and budget constraints common to most engineers, the prospect of extra work is a deterrent to the use of modeling. Therefore, any improvement in design procedures must also include an improvement in the efficiency of the time spent on the design. The use of a spreadsheet in performing these design calculations may help to increase this efficiency.

Spreadsheet Design Method

The spreadsheet provides an efficient environment for the tabular calculations necessary to perform a drainage system design. A general overview of the use of spreadsheets in water resources analysis is provided by Hancock and Heaney (1987). The design template has been constructed to emulate the FDOT Tabulation Form calculation procedure. A format has been developed such that pipe sizes, pipe slopes, and drops in manholes may be varied while constraints such as maximum and minimum velocities, hydraulic grade line, minimum cover, and design flow capacities are monitored. The spreadsheet automates calculations such as invert and crown elevations and partially filled pipe velocities, and estimates the total system cost according to a cost function or itemized costs for pipes and manholes. The automation of these calculations encourages a fine-tuning of the system design in order to develop a least-cost solution.

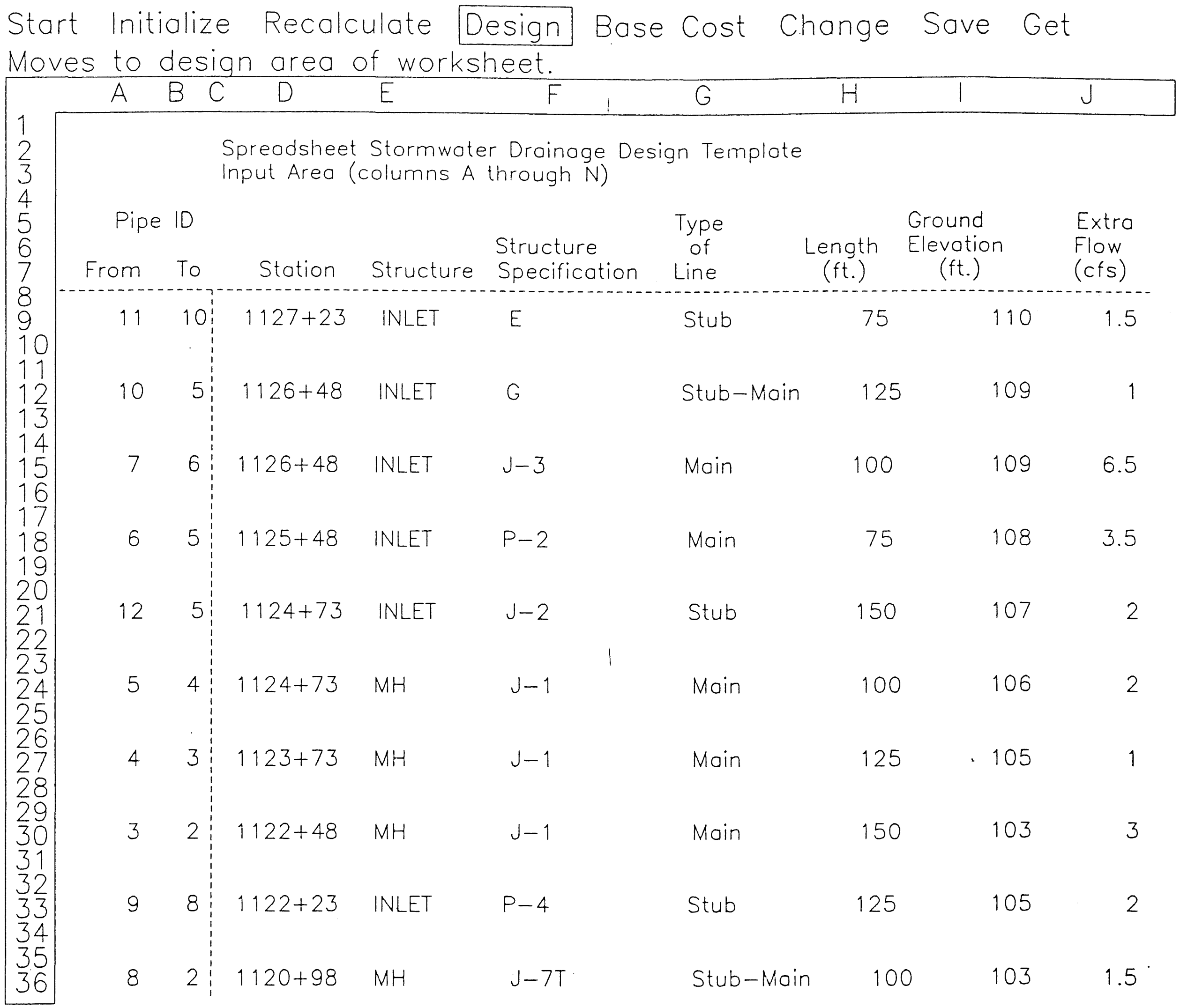
The spreadsheet design template has been developed as a tool with which practicing engineers may design stormwater drainage systems using familiar and easily understandable methods. The calculations follow conventional design procedures, but may be performed quickly and efficiently. This encourages an improvement in the design while not requiring increased time. The spreadsheet design algorithm is not rigid and may be modified to include individual strategies and engineering judgement. This feature is important since the solution method can be presented in a manner that is familiar to the engineer. Therefore, strategies and judgements that were developed while using the hand tabulation procedure may also be used in the spreadsheet method.

Problem Input

The design problem definition may be input into the spreadsheet in a very natural manner. Within the spreadsheet, the specifications for each pipe section are contained on one row as shown in Figure 2-3. Column headings define the position of each specification within the row. Every parameter has its own cell location where it may be referred to by its cell address (e.g. B21). Once the problem specifications have been entered into the input area of the spreadsheet, they may be used in calculations in other areas of the spreadsheet by this reference method. These parameters may be entered into the spreadsheet in any order and may be edited freely. Also shown in Figure 2-3 is the menu that has been created using the Lotus 1-2-3 macro language. The creation and use of this menu is discussed in a later section.

Heuristic Design Procedure

The heuristic design procedure consists of trying combinations of pipe sizes, pipe slopes, and manhole drops while monitoring problem constraints and system cost. As shown in Figure 2-4, pertinent design parameters are given for each pipe identification. Input data such as pipe length, Manning's n, type of line, design flow, and ground elevation are repeated in this area for easy reference during the design. The pipe capacity is calculated using Manning's equation for pipe size

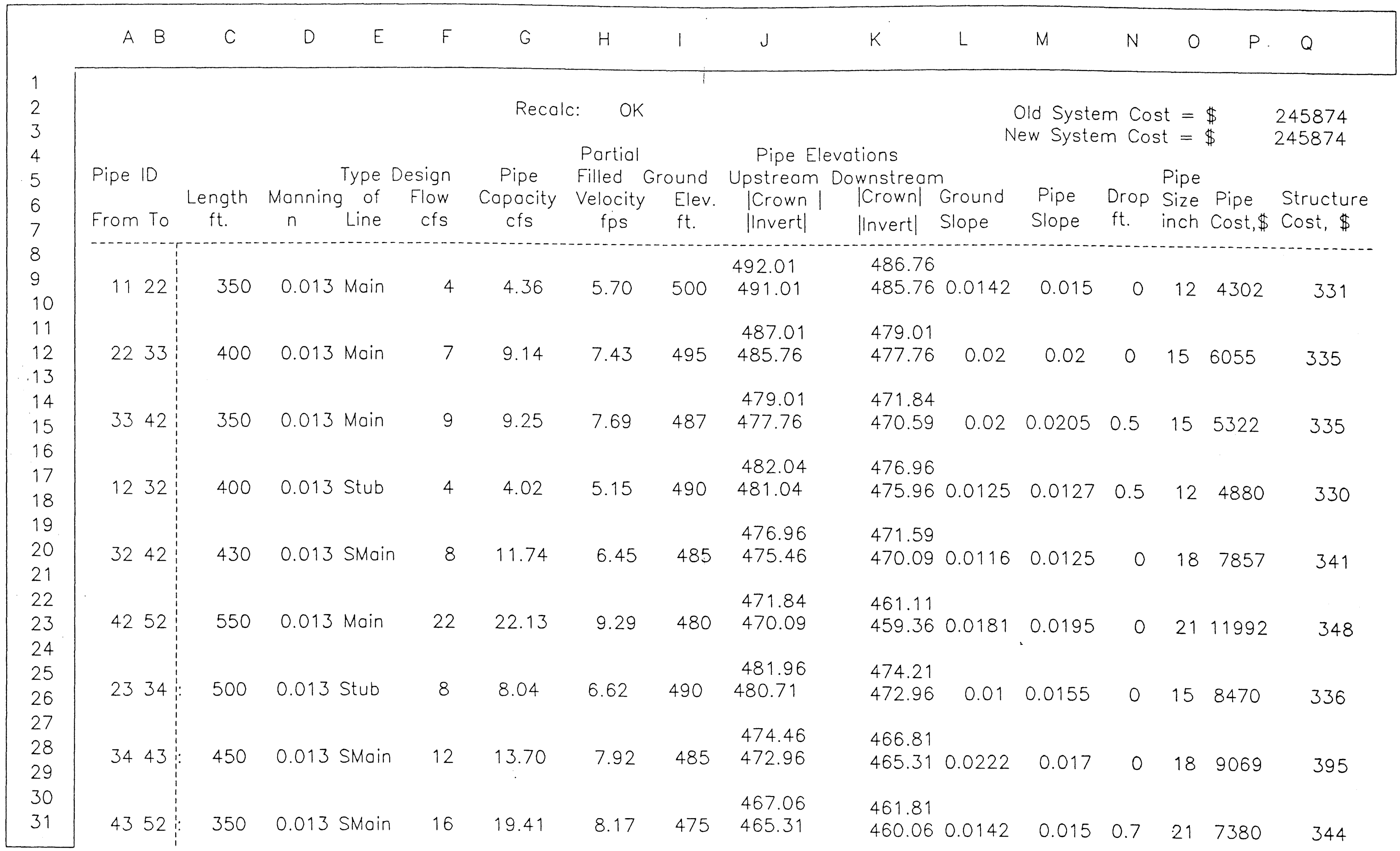


Moves to design area of worksheet.

Spreadsheet Stormwater Drainage Design Template
Input Area (columns A through N)

Pipe ID From	To	Station	Structure	Structure Specification	Type of Line	Length (ft.)	Ground Elevation (ft.)	Extra Flow (cfs)
11	10	1127+23	INLET	E	Stub	75	110	1.5
10	5	1126+48	INLET	G	Stub–Main	125	109	1
7	6	1126+48	INLET	J–3	Main	100	109	6.5
6	5	1125+48	INLET	P–2	Main	75	108	3.5
12	5	1124+73	INLET	J–2	Stub	150	107	2
5	4	1124+73	MH	J–1	Main	100	106	2
4	3	1123+73	MH	J–1	Main	125	105	1
3	2	1122+48	MH	J–1	Main	150	103	3
9	8	1122+23	INLET	P–4	Stub	125	105	2
8	2	1120+98	MH	J–7T	Stub–Main	100	103	1.5

Figure 2-3. Spreadsheet Drainage Design Input Area with Menu



A B C D E F G H I J K L M N O P Q

Recalc: OK

Old System Cost = $ 245874
New System Cost = $ 245874

Pipe ID From	To	Length ft.	Manning n	Type of Line	Design Flow cfs	Pipe Capacity cfs	Partial Filled Velocity fps	Ground Elev. ft.	Pipe Elevations Upstream |Crown| |Invert|	Downstream |Crown| |Invert|	Ground Slope	Pipe Slope	Drop ft.	Pipe Size inch	Pipe Cost,$	Structure Cost, $
11	22	350	0.013	Main	4	4.36	5.70	500	492.01 491.01	486.76 485.76	0.0142	0.015	0	12	4302	331
22	33	400	0.013	Main	7	9.14	7.43	495	487.01 485.76	479.01 477.76	0.02	0.02	0	15	6055	335
33	42	350	0.013	Main	9	9.25	7.69	487	479.01 477.76	471.84 470.59	0.02	0.0205	0.5	15	5322	335
12	32	400	0.013	Stub	4	4.02	5.15	490	482.04 481.04	476.96 475.96	0.0125	0.0127	0.5	12	4880	330
32	42	430	0.013	SMain	8	11.74	6.45	485	476.96 475.46	471.59 470.09	0.0116	0.0125	0	18	7857	341
42	52	550	0.013	Main	22	22.13	9.29	480	471.84 470.09	461.11 459.36	0.0181	0.0195	0	21	11992	348
23	34	500	0.013	Stub	8	8.04	6.62	490	481.96 480.71	474.21 472.96	0.01	0.0155	0	15	8470	336
34	43	450	0.013	SMain	12	13.70	7.92	485	474.46 472.96	466.81 465.31	0.0222	0.017	0	18	9069	395
43	52	350	0.013	SMain	16	19.41	8.17	475	467.06 465.31	461.81 460.06	0.0142	0.015	0.7	21	7380	344

Figure 2-4. Spreadsheet Template Design Area

and slope which have been entered. A description of the partially filled pipe velocity calculation is given in the following hydraulics section. The pipe crown and invert elevations are calculated from the pipe slope and length along with the pipe size and are referenced from downstream pipe elevations. The ground slope is provided to assist in using a strategy of minimizing excavation by keeping the pipe slope similar to the ground slope if desired. An estimate of the installed pipe cost and structure cost for each pipe section is also given and is automatically updated according to any design parameter change. These cost estimates may be used as a heuristic function to direct the design procedure towards a least-cost solution.

The design area in the spreadsheet template is intended to contain the most important information needed in proceeding through the system design. In order to keep all of this information on the screen while moving from pipe to pipe, the Lotus 1-2-3 ability to create row and column titles is used. Titles will remain at the top of the screen while the cursor is moved up or down to view other pipe sections. Similarly, the pipe identification columns will remain at the left of the screen while the cursor is moved to view other pipe section parameters to the left or right of those shown. With this feature, the column headings and pipe identifications are always known for the parameters being viewed.

The design procedure begins at the downstream end of the system and proceeds upstream. For each section, the pipe size and slope are varied manually while the flow capacity is checked against the design flow capacity. The hydraulic grade line is calculated beginning at the downstream end of the pipe network at a user given elevation. Only the

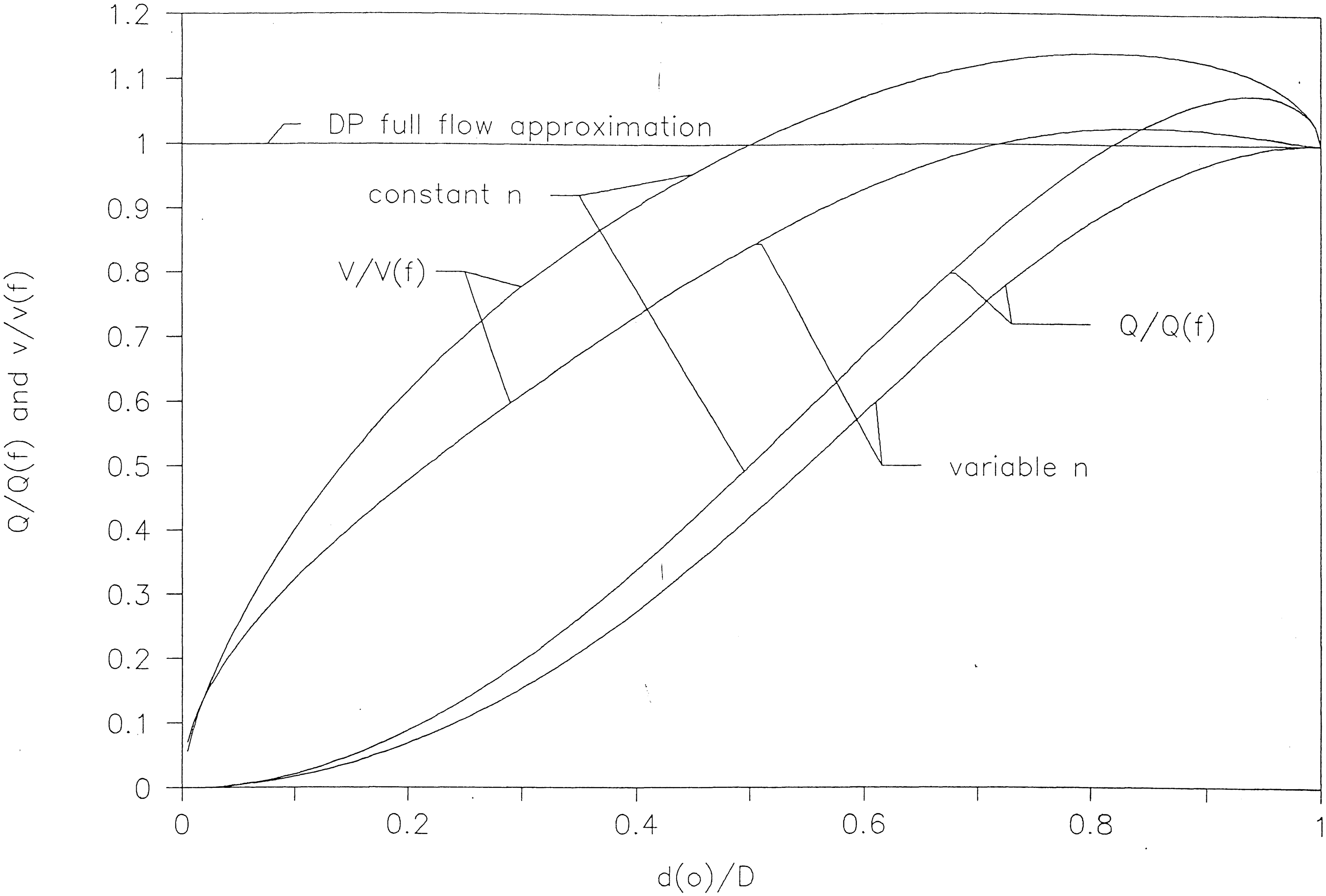


Figure 2-5. Hydraulic Elements of a Circular Pipe for Constant and Variable Roughness and Dynamic Programming Full Flow Approximation

solutions, his equations were approximated. Q/Q(f) and v/v(f) were calculated for small increments of d(o)/D from 0 to 1 and tabulated. A Lotus lookup table function was used to retrieve the value of v/v(f) given the corresponding value of Q/Q(f). A similar tabular solution method is also used to find values of v/v(f) for the constant roughness approach in order to save on calculation time.

The ratio of Q/Q(f) is known in the design problem. Therefore, the v/v(f) ratio may be obtained from the lookup table and multiplied by the full flow velocity to get the partially filled pipe velocity. The increments in the lookup table are sufficiently small so that the maximum error obtained is less than 1%. This error could be further reduced by decreasing the increment size in the table without significantly increasing the calculation time of the lookup function.

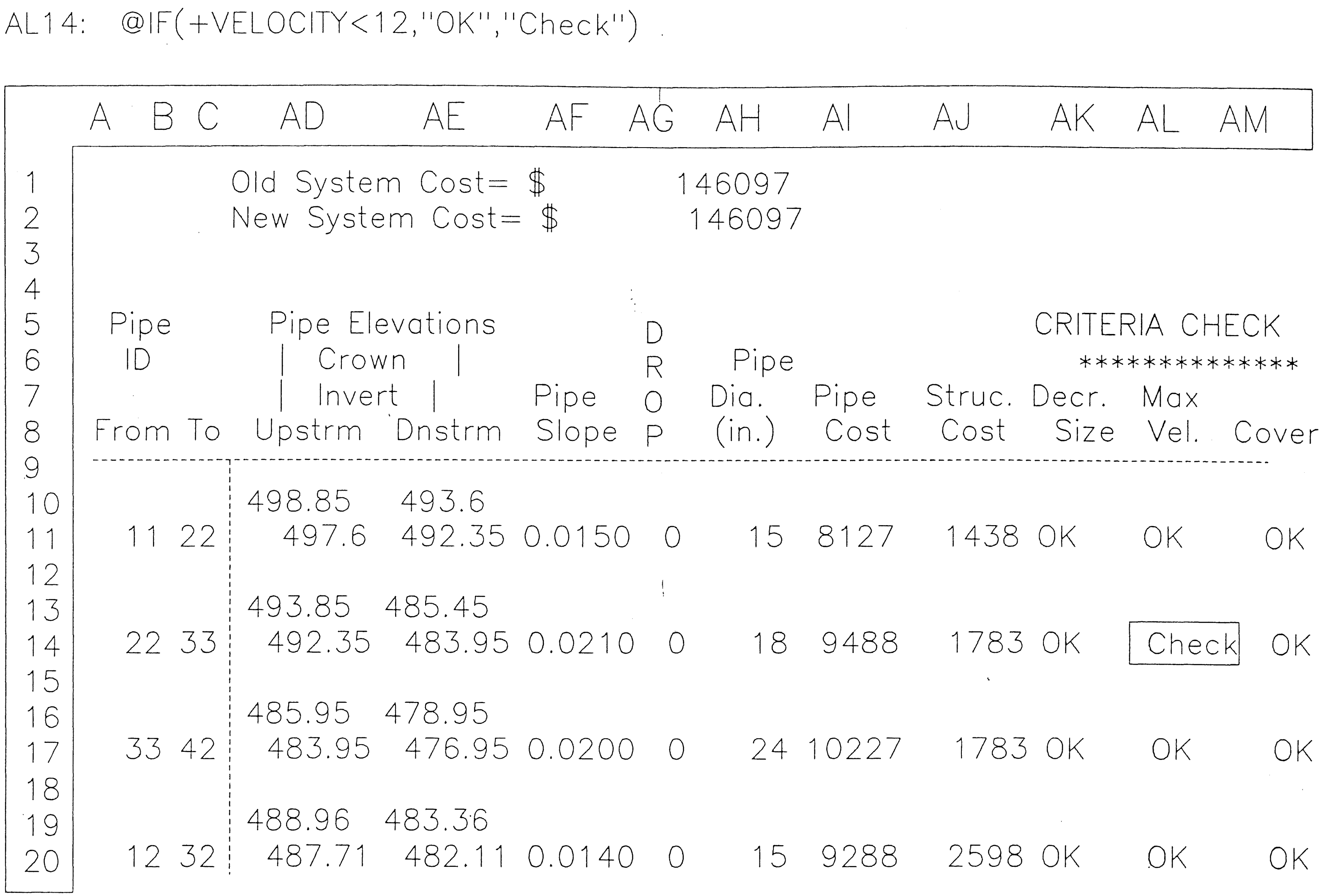
Also shown in Figure 2-5 is the approximation used by the dynamic programming algorithms for the partially full flow calculations. This approximation becomes a straight line on the hydraulic elements figure since the velocity is always assumed to be the velocity of a pipe flowing full regardless of the depth of flow in the pipe. As can be seen, this approximation can produce large errors when the flow in the pipe is other than the full flow capacity of the pipe.

Overview of Design Template Features

Menus/Macros. The Lotus 1-2-3 spreadsheet includes the powerful capability of an internal macro programming language. This language can be used as a keystroke recorder of repetitive spreadsheet commands, or can execute programmed code which is similar in many ways to BASIC. The macro language includes the ability to create subroutines, conditional branches, for loops, and many other structures which are found in conventional programming languages. Several references are available that describe the format of constructing macro programs and give example programs for specific tasks (Lotus Development Corp., 1986; Simpson, 1986). The ability to create menus is useful when a template is created that contains several macro programs. Menus provide the capability to incorporate all macros into a single macro from which all other macros are called. Each macro is given a name on the menu. When that name is chosen from the menu, the macro is executed. Also included is the ability to provide a short description of the macro as the cursor is moved onto its name in the menu, much like the normal Lotus command menus (see Figure 2-3). The menu capability becomes especially handy when a template is to be used by others.

Recalculation. A problem that does arise in large spreadsheet templates is the recalculation mode. In a spreadsheet program, every time the return key is pressed (when entering a value or a label, etc.) the entire spreadsheet is recalculated. This becomes a problem when the spreadsheet is very large and contains many calculations. A solution to this problem is to change the spreadsheet to manual recalculation mode. In this mode, the spreadsheet is not recalculated until the user presses the "F9" key. This speeds the entry of values and labels; however, during the iterative design procedure, the spreadsheet needs to be recalculated often to display the results of a design change. To solve this problem, the {recalc} macro command is used. This macro command allows only a portion of the spreadsheet to be recalculated at a time. Therefore, when a design change is made, only the design portion of the spreadsheet must be recalculated. For very large pipe systems, this recalculation time is reduced even further by recalculating only one pipe at a time. This ability is useful to quickly verify that a design change meets the specified criteria. The recalculation of the entire design portion may then be performed only periodically for cost estimate updates. The last problem with the recalculation of the design portion is the verification that a design change in a downstream section of the system has been reflected in all necessary upstream segments. This verification is performed with a recalculation indicator. This indicator, which is located at the top of the design screen as shown in Figure 2-4, compares the summation of all numbers contained in the design section from consecutive recalculations. If the summation from two consecutive recalculations is within a small value, then the indicator displays "ok." Otherwise the indicator displays "check." In this manner the user is assured that all design changes have been reflected in all upstream portions of the system.

Automated Criteria Checks. The criteria checks in the system design are most often performed manually. Each time the design is changed, the velocity in a pipe is visually compared to the minimum and maximum constraints. The provision is included, however, for automated criteria checks. Conditional statements are entered into cells to the right of the design area of the template which compares calculated values with specified criteria. For example, the conditional statement compares the calculated velocity with the specified limits. If the velocity is outside those limits, then the cell displays "check" (see Figure 2-6). Otherwise the cell displays "ok." These automated checks are most useful as a final verification once a system design is complete. These checks are included for velocity, hydraulic grade line and cover, and flow capacity constraints.



```
AL14:  @IF(+VELOCITY<12,"OK","Check")

      A  B  C   AD       AE      AF    AG    AH     AI      AJ     AK    AL     AM

 1               Old System Cost= $            146097
 2               New System Cost= $             146097
 3
 4
 5    Pipe      Pipe Elevations        D                           CRITERIA CHECK
 6     ID       |  Crown    |          R    Pipe                      *************
 7              |  Invert  |     Pipe  O    Dia.   Pipe   Struc.  Decr.   Max
 8   From To  Upstrm  Dnstrm    Slope  P    (in.)  Cost    Cost    Size   Vel.   Cover
 9   -------------------------------------------------------------------------------
10          | 498.85   493.6
11    11 22 |   497.6  492.35 0.0150  0     15   8127    1438  OK     OK       OK
12          |
13          | 493.85  485.45
14    22 33 |  492.35  483.95 0.0210  0     18   9488    1783  OK    Check     OK
15          |
16          | 485.95  478.95
17    33 42 |  483.95  476.95 0.0200  0     24  10227    1783  OK     OK       OK
18          |
19          | 488.96  483.36
20    12 32 |  487.71  482.11 0.0140  0     15   9288    2598  OK     OK       OK
```

Figure 2-6. Example of Conditional Statement Used in Automated Criteria Check

Conclusions

The tabulation procedure that is presently used by the FDOT for stormwater drainage design calculations is tedious to perform manually. The repetitive calculations that this type of procedure require are subject to error and are not conducive to finding the least-cost solution for the design problem. The procedure does not explicitly consider system cost as a design criteria and so the engineer has no basis to judge a design in terms of cost if desired.

The spreadsheet method applies very few procedural changes from the tabulation procedure. The spreadsheet acts as a tool that allows the engineer to more efficiently perform the necessary calculations for a drainage design. The repetitive calculations are automated so as to minimize errors, but are based on the same procedure presently used so as to be easily understood. The ease with which the calculations are performed encourages the engineer to find a better solution through many iterations. The spreadsheet template additionally provides an automated cost estimate which may be used as a heuristic function in finding a least-cost design. This feature enables the engineer to acquire a good feel for the trade-offs between pipe size and excavation which exist in this design problem.

More importantly, however, the spreadsheet allows drainage engineers who are familiar with the tabulation procedure to employ individual strategies already acquired. The spreadsheet procedure does not employ a rigid algorithm. This design template simply provides an efficient environment with which the engineer may apply his or her own knowledge and judgement.

CHAPTER 3
COMPARISON OF SPREADSHEET TECHNIQUE WITH DYNAMIC PROGRAMMING

Literature Review

Research dealing with the least cost design of stormwater drainage systems using dynamic programming algorithms has been conducted for many years. Early work was done by Zepp and Leary (1969), Meredith (1972), and Merritt and Bogan (1973). More recently, Robinson and Labadie (1981) describe a general dynamic programming algorithm called CSUDP which was developed at Colorado State University. This computer code has been applied to the urban drainage design problem in conjunction with a subroutine called SEWER. Robinson and Labadie provide solutions to three example problems using CSUDP-SEWER. The last of these problems was originally solved by Mays and Wenzel (1976) using discrete differential dynamic programming (DDDP). In both of these papers the cost of the drainage system was determined by a cost function used in earlier work by Meredith (1972). The DDDP algorithm used by Mays and Wenzel has since been integrated into the Illinois Least-Cost Sewer System Design Model (ILSD) (Yen et al. 1984).

More recently electronic spreadsheets have been used to address this problem. Brown and Koussis (1987) showed that the spreadsheet could be used to efficiently perform the hydraulic calculations needed to design a stormwater drainage system. The spreadsheet was used to implement their method which was shown to provide accurate designs while using less complex computations than large mainframe programs. Their method was aimed at improving over rational formula design methods while providing a method which will be accepted by potential users. Miles et al. (1987) used the spreadsheet to perform detailed cost estimates of stormwater drainage systems. The spreadsheet was shown to be useful in providing automatic in-depth cost estimates using a computerized database of drainage items. These cost estimates were found useful for system analysis during the design procedure.

Description of Dynamic Programming Algorithm

A brief description of the mechanics of applying a dynamic programming algorithm to the optimization of stormwater systems is provided in order to familiarize the reader with this process. Many textbooks are available for a detailed description of dynamic programming algorithms (e.g. Smith et al., 1983; Hillier and Lieberman, 1986) and the authors mentioned in the literature review have written many papers dealing specifically with the application of dynamic programming to the stormwater drainage problem. These references may be consulted for a more in-depth understanding of this algorithm.

Dynamic programming is a method of solving multistage problems, or problems where the outcome of a decision at one stage affects the decision at the next stage. At each stage in the problem there are discrete alternatives which are called states (see Figure 3-1). In the vertical alignment problem each stage is defined by a pipe and its downstream manhole, and each state represents an alternative elevation at which the pipe may be placed at that manhole. Figure 3-1 is similar to Figure 2-1 except that only the pipe crowns are shown and each alternative pipe placement is shown. The distance between each of the alternative states is called delta x, or the state increment size.

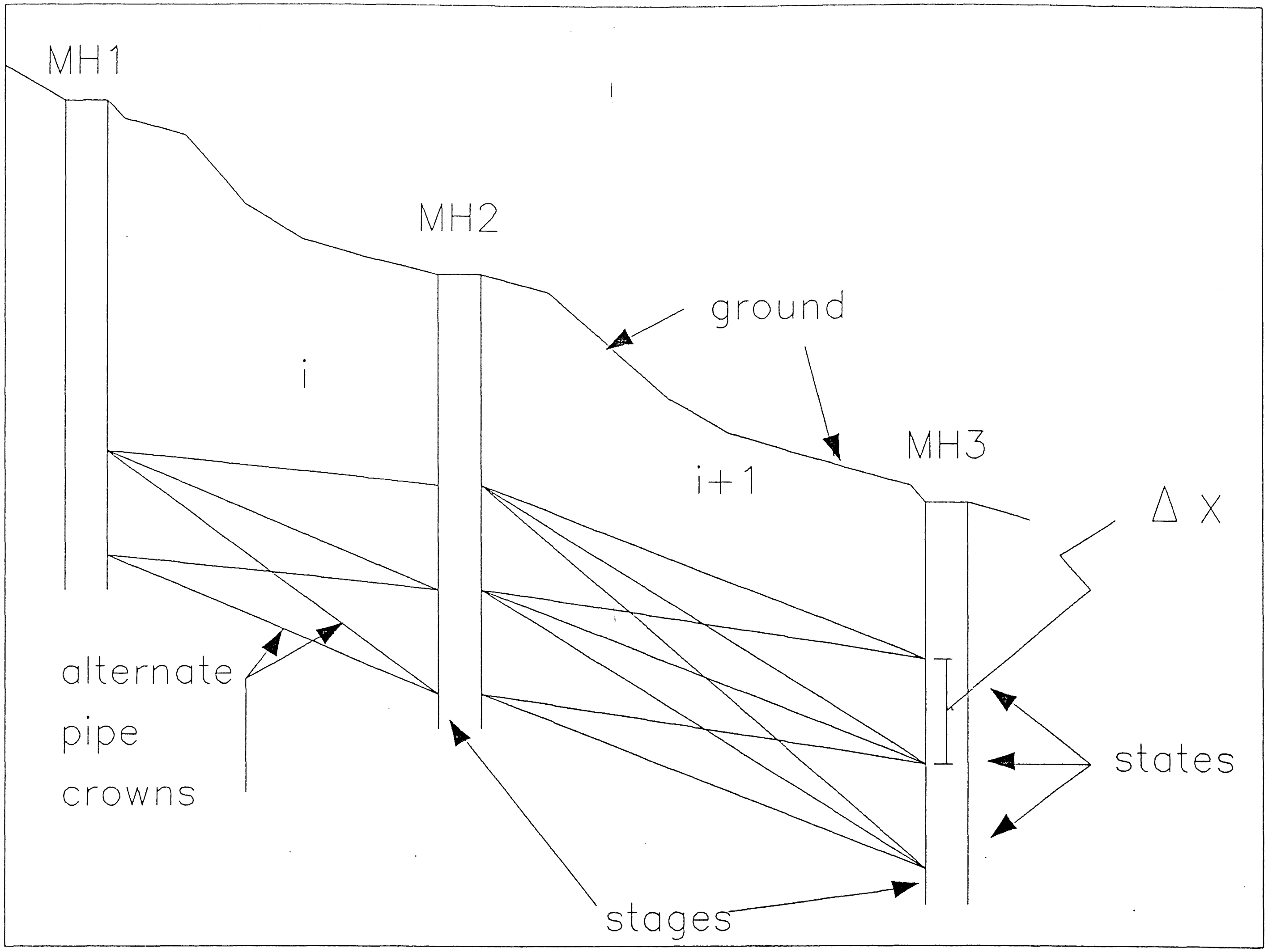


Figure 3-1. Illustration of Dynamic Programming Approach to the Vertical Alignment Problem

The elevation at which a pipe is placed affects the optimal elevation of the downstream pipe. At each stage of the solution procedure the cost of each state is evaluated using a cost function, and the least-cost solution path to arrive at that state is saved recursively. Since only the least-cost solution path to arrive at that state is saved, the algorithm is relieved of totally enumerating all possible solutions. Once this evaluation is performed for all stages, then the algorithm traces back through the system to find the final least-cost design.

The cost function (or stage return function) estimates the cost at each state by estimating the installed costs of the pipes and manholes. In order to determine the pipe costs, the pipe diameter (a decision variable) must be known. This decision variable is found with the state transformation equation (Manning's equation) given the desired flow capacity and by calculating the pipe slope from the known upstream and downstream crown elevations. Additional constraints of minimum and maximum velocities in the pipe and a minimum cover above the crown of the pipe must also be checked. These constraints are met by assigning large penalty costs to all solutions in which these constraints are violated. This strategy allows a solution in which a constraint has been violated to be chosen only if no other feasible solution is found.

Once the algorithm has found the least-cost solution for the originally specified delta x, then the delta x is decreased in order to increase the accuracy of the solution. The new (and more densely spaced) state elevations are placed to span the previously found least-cost states and the problem is resolved. This process is continued until the desired accuracy is obtained.

Mathematically this problem may be formulated as:

$$\text{minimize } Z = \overset{N}{\underset{i=1}{\quad}} f_i(x_i, d_i, x_{i-1})$$

subject to: $d_i = [(2.16*n*Q_i)/(((x_{i-1}-x_i)/L_i)\hat{\ }.5)]\hat{\ }0.375$

$x_i \quad X_i$

$d_i \quad D$

where:

$f_i()$ = cost function for pipe section i,

d_i = diameter of pipe section i,

$((x_{i-1}-x_i)/L_i)$ = slope of the pipe section i,

x_i = elevation of pipe crown at end of section i,

L_i = length of pipe section i,

Q_i = rate of flow in pipe section i,

N = number of stages in system,

n = Manning's roughness coefficient,

D = set of discrete values of pipe diameters,

X_i = set of discrete values of pipe elevations at end of section i

Other constraints such as minimum and maximum velocities are problem specific and may be of the form: $h_i = h_i(x_i, u_i) <=>$ constant.

Example Problem

A design problem which has been previously solved using dynamic programming techniques has also been solved using the spreadsheet method. This problem was originally presented by Mays and Wenzel (1976) in a description of their discrete differential dynamic programming approach. Robinson and Labadie (1981) have also solved the Mays and Wenzel problem using their CSUDP algorithm. The cost estimates in each of these papers were performed using the following pipe and manhole cost functions which were used earlier by Meredith (1972):

$$C_p = \begin{array}{l} 10.98*d + 0.8*H - 5.98 \text{ if } d \leq 3 \text{ ft and } H \leq 10 \text{ ft.} \\ 5.94*d + 1.166*H + 0.504*H*d - 9.64 \text{ if } d \leq 3 \text{ ft and } H \geq 10 \text{ ft.} \\ 30.0*d + 4.9*H - 105.9 \text{ if } d > 3 \text{ ft.} \end{array}$$

$$C_m = 250 + h\hat{\ }2$$

where : C_p = installed pipe cost (\$/linear foot of pipe)

C_m = installed manhole cost (\$)

d = pipe diameter (ft.)

H = average invert depth below the ground surface (ft.), and

h = manhole depth (ft.)

The spreadsheet method cost estimates have also used Meredith's functions so that direct comparisons could be made with the previously published results.

Mays and Wenzel (1976) presented a discrete differential dynamic programming (DDDP) algorithm which was used to solve the twenty pipe drainage system shown in Figure 3-2. This problem was constrained to have a maximum velocity of 8 fps, a minimum velocity of 2 fps, and a minimum cover of 8 ft. Robinson and Labadie (1981) also solved this problem and compared their results to the original solution. The original solution derived by Mays and Wenzel using their DDDP algorithm and Meredith's cost functions gave a least cost solution of $265,355. Because Mays and Wenzel only gave the pipe sizes and elevations and did not give resulting velocities, their result was replicated on the spreadsheet template. This spreadsheet duplication gave a cost estimate of $265,775, a difference of less than 0.2% from the original Mays and Wenzel solution, and so it is assumed to be a very close reproduction. The spreadsheet reproduction did reveal, however, that the Mays and Wenzel solution produced velocities which exceeded the defined constraint of 8 fps in 7 out of 20 pipes in the system (see Table 3-1). These exceedences result from the assumption of full-flowing pipes in the original solution algorithm.

Robinson and Labadie presented several solutions to the Mays and Wenzel problem allowing different assumptions. Their best result of $274,463 was obtained with the assumptions of no drops allowed in manholes and full-flowing pipes. In assuming full-flowing pipes, however, Robinson and Labadie also introduced inaccuracies in their velocity calculations. In calculating the velocity in the pipe, the

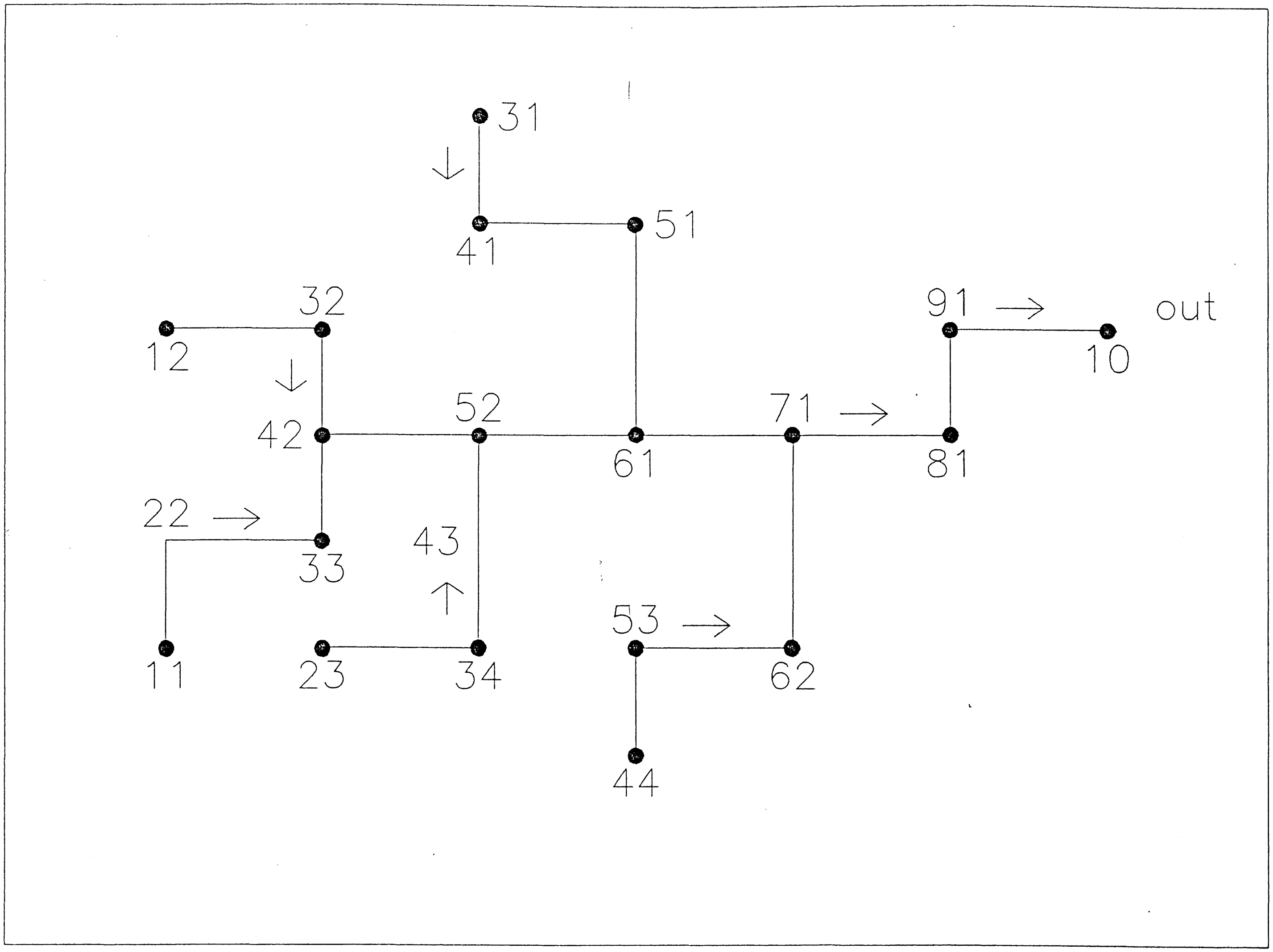


Figure 3-2. System Layout of Twenty Pipe Problem Presented by Mays and Wenzel (1976)

Table 3-1. Replication of Mays and Wenzel Solution

Recalc: OK

New System Cost = $265775

Pipe ID From	To	Manning n	Type of Line	Design Pipe Capac. cfs	Full Flow Capac. cfs	Vel. fps	Ground Elev.	Crown Elevation Invert Elevation Upstream	Downstr.	Pipe Slope	Drop in MH ft.	Pipe Diam. in.	Pipe Cost $	Structure Cost $	velocity <8 fps
11	22	0.013	Main	4	4.27	4.84	500	492.00 491.00	487.00 486.00	0.014	0.25	12	4411	350	yes
22	33	0.013	Main	7	9.14	7.43	495	487.00 485.75	479.00 477.75	0.020	0	15	6058	336	yes
33	42	0.013	Main	9	9.14	7.56	487	479.00 477.75	472.00 470.75	0.020	0.75	15	5301	336	yes
12	32	0.013	Stub	4	7.05	5.29	490	482.00 480.75	477.24 475.99	0.012	0.5	15	6020	336	yes
32	42	0.013	SMain	8	11.31	6.27	485	476.99 475.49	472.00 470.50	0.012	0.5	18	7781	340	yes
42	52	0.013	Main	22	21.38	8.89	480	471.75 470.00	461.74 459.99	0.018	0.5	21	11685	350	no
23	34	0.013	Stub	8	10.50	5.92	490	482.00 480.50	477.00 475.50	0.010	0	18	9047	340	yes
34	43	0.013	SMain	12	15.65	8.84	485	477.00 475.50	467.01 465.51	0.022	0.25	18	8141	340	no
43	52	0.013	SMain	16	18.95	8.01	475	467.01 465.26	462.00 460.25	0.014	0.76	21	7362	345	yes
52	61	0.013	Main	44	43.99	8.96	470	461.99 459.49	456.24 453.74	0.012	4.84	30	15809	360	no
31	41	0.013	Stub	9	9.14	7.56	485	477.01 475.76	467.01 465.76	0.020	0.5	15	7571	335	yes
41	51	0.013	SMain	16	18.95	8.01	475	467.01 465.26	462.00 460.25	0.014	0.25	21	7362	345	yes
51	61	0.013	SMain	20	19.54	8.12	470	461.75 460.00	456.43 454.68	0.015	5.78	21	7547	350	no
61	71	0.013	Main	71	95.45	9.84	465	452.40 448.90	447.00 443.50	0.009	0.38	42	40032	509	no
44	53	0.013	Stub	4	7.08	5.29	468	460.04 458.79	455.24 453.99	0.012	0.27	15	6173	335	yes
53	62	0.013	SMain	6	6.46	5.40	464	454.97 453.72	451.97 450.72	0.010	0.25	15	4671	356	yes
62	71	0.013	SMain	9	12.52	6.97	460	451.97 450.47	447.00 445.50	0.014	2.38	18	6336	341	yes
71	81	0.013	Main	87	95.98	10.22	455	446.62 443.12	442.98 439.48	0.009	0	42	22572	391	no
81	91	0.013	Main	89	88.86	9.24	451	442.98 439.48	439.08 435.58	0.008	0.5	42	28877	383	no
91	10	0.013	Main	94	94.19	7.54	448	439.08 435.08	436.50 432.50	0.004	0	48	45827	417	yes
10	10						445	432.50 432.50	432.50 432.50					417	
											Total		258580	7195	

design flow in the pipe was divided by the cross sectional area of the pipe to obtain a velocity, even if the actual flow capacity of the pipe was much larger. Therefore, if a pipe were flowing half-full, the full flow velocity calculated by Robinson and Labadie would be about one-half of the true full flow velocity. This inaccuracy is in addition to the previously mentioned assumption that the partially full flow velocity is always equal to the full flow velocity. Since the velocities calculated by Robinson and Labadie were inaccurate, their solution was also replicated on the spreadsheet template (see Table 3-2). A cost estimate of $275,218 was calculated with the spreadsheet replication as compared to $274,463 for Robinson and Labadie's original solution, a difference of less than 0.3%. One pipe in this solution showed a velocity of 12.2 fps in the replication where the velocity had been calculated as 7.38 fps by Robinson and Labadie. The replication showed that a total of eight out of twenty pipes exceeded the velocity constraint of 8 fps defined in the problem.

The spreadsheet design method shown in Table 3-3 produced a solution to this problem which meets all velocity and cover constraints. Using Meredith's cost function, an estimate of $254,406 was produced for this solution. This represents a 4.1% reduction over the Mays and Wenzel solution and an 7.3% reduction over the Robinson and Labadie solution. In this solution, manhole drops of 7.5 and 4.9 ft are used. It is unusual that such a design would be used although the original Mays and Wenzel solution also had manhole drops of 5.8 and 4.8 ft. The problem was solved again in order to eliminate these drops. In this solution, the velocity constraint was relaxed to 12 fps and pipe crowns were matched at manholes. As shown in Table 3-4, the cost of this

Table 3-2. Replication of Robinson and Labadie Solution

Recalc: OK New System Cost = $275218

Pipe ID From	To	Manning n	Type of Line	Design Pipe Capac. cfs	Full Flow Capac. cfs	Vel. fps	Ground Elev.	Crown Elevation Invert Elevation Upstream	Downstr.	Pipe Slope	Drop in MH ft.	Pipe Diam. in.	Pipe Cost $	Structure Cost $	velocity <8 fps
								492.03	487.03						
11	22	0.013	Main	4	4.26	5.56	500	491.03	486.03	0.014	0.25	12	4261	330	yes
								487.03	479.03						
22	33	0.013	Main	7	9.14	7.43	495	485.78	477.78	0.020	0	15	6048	335	yes
								479.03	472.03						
33	42	0.013	Main	9	9.14	7.56	487	477.78	470.78	0.020	0.75	15	5292	335	yes
								482.05	476.97						
12	32	0.013	Stub	4	4.02	5.15	490	481.05	475.97	0.013	0.5	12	4875	330	yes
								476.97	472.03						
32	42	0.013	SMain	8	11.26	6.24	485	475.47	470.53	0.012	0.5	18	7778	341	yes
								472.03	462.02						
42	52	0.013	Main	22	30.52	9.56	480	470.03	460.02	0.018	1	24	13178	349	no
								482.02	477.02						
23	34	0.013	Stub	8	10.50	5.92	490	480.52	475.52	0.010	0	18	9039	340	yes
								477.02	467.03						
34	43	0.013	SMain	12	15.65	8.84	485	475.52	465.53	0.022	0.25	18	8133	340	no
								467.03	462.02						
43	52	0.013	SMain	16	18.95	8.01	475	465.28	460.27	0.014	1.25	21	7356	345	yes
								462.02	457.02						
52	61	0.013	Main	44	66.70	9.07	470	459.02	454.02	0.010	0.5	36	18792	371	no
								477.03	467.03						
31	41	0.013	Stub	9	9.14	7.56	485	475.78	465.78	0.020	0.5	15	7561	335	yes
								467.03	462.03						
41	51	0.013	SMain	16	18.95	8.01	475	465.28	460.28	0.014	0.25	21	7355	344	yes
								462.03	457.02						
51	61	0.013	SMain	20	27.05	8.52	470	460.03	455.02	0.014	1.5	24	8387	350	no
								457.02	447.00						
61	71	0.013	Main	71	130.02	12.29	465	453.52	443.50	0.017	0.5	42	33241	382	no
								460.28	455.20						
44	53	0.013	Stub	4	7.28	5.42	468	459.03	453.95	0.013	0.25	15	6143	331	yes
								454.95	452.01						
53	62	0.013	SMain	6	6.39	5.34	464	453.70	450.76	0.010	0.25	15	4670	356	yes
								452.01	447.00						
62	71	0.013	SMain	9	12.56	6.98	460	450.51	445.50	0.014	2.5	18	6331	340	yes
								447.00	443.00						
71	81	0.013	Main	87	143.64	10.71	455	443.00	439.00	0.010	0	48	29160	394	no
								443.00	440.00						
81	91	0.013	Main	89	111.27	8.93	451	439.00	436.00	0.006	0	48	36450	394	no
								440.00	437.00						
91	10	0.013	Main	94	101.57	8.29	448	436.00	433.00	0.005	0	48	43740	394	no
								433.00	433.00						
10	10						445	433.00	433.00					394	
											Total		267789	7429	

Table 3-3. Spreadsheet Solution with All Constraints Met

Recalc: OK New System Cost = $254407

Pipe ID		Manning n	Type of Line	Design Pipe Capac. cfs	Full Flow Capac. cfs	Vel. fps	Ground Elev.	Crown Elevation Invert Elevation		Pipe Slope	Drop in MH ft.	Pipe Diam. in.	Pipe Cost $	Structure Cost $	velocity <8 fps
From	To							Upstream	Downstr.						
								492.03	487.48						
11	22	0.013	Main	4	4.06	5.25	500	491.03	486.48	0.013	0.7	12	4199	330	yes
								487.03	482.23						
22	33	0.013	Main	7	7.08	5.84	495	485.78	480.98	0.012	3.2	15	5536	335	yes
								479.03	472.03						
33	42	0.013	Main	9	9.14	7.56	487	477.78	470.78	0.020	1.6	15	5292	335	yes
								482.05	476.85						
12	32	0.013	Stub	4	4.06	5.25	490	481.05	475.85	0.013	0	12	4898	330	yes
								477.10	470.43						
32	42	0.013	SMain	8	8.04	6.62	485	475.85	469.18	0.016	0	15	6766	334	yes
								471.18	465.96						
42	52	0.013	Main	22	22.05	7.06	480	469.18	463.96	0.010	4.9	24	12499	367	yes
								482.05	474.30						
23	34	0.013	Stub	8	8.04	6.62	490	480.80	473.05	0.016	0	15	8393	335	yes
								474.55	468.61						
34	43	0.013	SMain	12	12.07	6.89	485	473.05	467.11	0.013	1.8	18	8293	393	yes
								467.06	463.21						
43	52	0.013	SMain	16	16.62	7.06	475	465.31	461.46	0.011	2.4	21	7186	344	yes
								462.06	459.81						
52	61	0.013	Main	44	44.74	6.43	470	459.06	456.81	0.005	3.3	36	17308	370	yes
								477.08	467.33						
31	41	0.013	Stub	9	9.02	7.39	485	475.83	466.08	0.020	0.8	15	7490	334	yes
								467.03	463.36						
41	51	0.013	SMain	16	16.24	6.85	475	465.28	461.61	0.011	1.6	21	7168	344	yes
								462.01	459.21						
51	61	0.013	SMain	20	20.23	6.53	470	460.01	457.21	0.008	3.7	24	8084	350	yes
								457.01	454.01						
61	71	0.013	Main	71	71.14	7.43	465	453.51	450.51	0.005	7.5	42	22965	382	yes
								460.06	456.06						
44	53	0.013	Stub	4	6.46	4.97	468	458.81	454.81	0.010	0	15	6040	335	yes
								456.06	453.36						
53	62	0.013	SMain	6	6.13	5.09	464	454.81	452.11	0.009	1.3	15	4374	335	yes
								452.06	445.06						
62	71	0.013	SMain	9	9.14	7.56	460	450.81	443.81	0.020	0.8	15	5633	335	yes
								447.01	445.41						
71	81	0.013	Main	87	90.85	7.40	455	443.01	441.41	0.004	2.4	48	26798	394	yes
								443.01	441.08						
81	91	0.013	Main	89	89.13	7.13	451	439.01	437.08	0.004	1	48	35121	394	yes
								440.08	437.50						
91	10	0.013	Main	94	94.19	7.54	448	436.08	433.50	0.004	1	48	42887	392	yes
								432.50	432.50						
10	10						445	432.50	432.50					406	
											Total		246934	7473	

Table 3-4. Spreadsheet Solution with no Drops in Manholes and Velocity < 12 fps

Recalc: OK　　New System Cost = $245874

Pipe ID From	To	Manning n	Type of Line	Design Pipe Capac. cfs	Full Flow Capac. cfs	Vel. fps	Ground Elev.	Crown Elevation Invert Elevation Upstream	Crown Elevation Invert Elevation Downstr.	Pipe Slope	Drop in MH ft.	Pipe Diam. in.	Pipe Cost $	Structure Cost $	velocity <12 fps
11	22	0.013	Main	4	4.36	5.70	500	492.01 491.01	486.76 485.76	0.015	0	12	4302	331	yes
22	33	0.013	Main	7	9.14	7.43	495	487.01 485.76	479.01 477.76	0.020	0	15	6055	335	yes
33	42	0.013	Main	9	9.25	7.69	487	479.01 477.76	471.84 470.59	0.021	0.5	15	5322	335	yes
12	32	0.013	Stub	4	4.02	5.15	490	482.04 481.04	476.96 475.96	0.013	0.5	12	4880	330	yes
32	42	0.013	SMain	8	11.74	6.45	485	476.96 475.46	471.59 470.09	0.013	0	18	7857	341	yes
42	52	0.013	Main	22	22.13	9.29	480	471.84 470.09	461.11 459.36	0.020	0	21	11992	348	yes
23	34	0.013	Stub	8	8.04	6.62	490	481.96 480.71	474.21 472.96	0.016	0	15	8470	336	yes
34	43	0.013	SMain	12	13.70	7.92	485	474.46 472.96	466.81 465.31	0.017	0	18	9069	395	yes
43	52	0.013	SMain	16	19.41	8.17	475	467.06 465.31	461.81 460.06	0.015	0.7	21	7380	344	yes
52	61	0.013	Main	44	44.93	9.32	470	461.86 459.36	455.86 453.36	0.012	0	30	16118	363	yes
31	41	0.013	Stub	9	9.14	7.56	485	476.96 475.71	466.96 465.71	0.020	0.5	15	7589	336	yes
41	51	0.013	SMain	16	18.75	7.94	475	466.96 465.21	462.06 460.31	0.014	0	21	7359	346	yes
51	61	0.013	SMain	20	20.66	8.77	470	462.06 460.31	456.11 454.36	0.017	1	21	7551	344	yes
61	71	0.013	Main	71	83.04	11.97	465	456.36 453.36	447.06 444.06	0.016	0.5	36	23049	385	yes
44	53	0.013	Stub	4	4.06	5.25	468	460.01 459.01	454.81 453.81	0.013	0	12	5070	331	yes
53	62	0.013	SMain	6	6.46	5.40	464	455.06 453.81	452.06 450.81	0.010	0	15	4650	354	yes
62	71	0.013	SMain	9	9.29	7.73	460	452.06 450.81	444.81 443.56	0.021	0	15	5710	335	yes
71	81	0.013	Main	87	100.61	10.67	455	447.06 443.56	443.06 439.56	0.010	0	42	22062	381	yes
81	91	0.013	Main	89	89.99	9.48	451	443.06 439.56	439.06 435.56	0.008	0	42	28803	381	yes
91	10	0.013	Main	94	102.58	8.37	448	439.56 435.56	436.50 432.50	0.005	0	48	45122	405	yes
10	10						445	432.50 432.50	432.50 432.50					406	
											Total		238411	7463	

solution was $245,873. This represents a 7.3% and a 10.4% decrease over the Mays and Wenzel and the Robinson and Labadie solutions, respectively, which both also included velocities of over 10 fps in their solutions.

How Can Spreadsheet Designs Be Better Than Optimal?

The spreadsheet design method has been shown to be better than the optimal designs found by both dynamic programming algorithms. This may be attributed partially to using a more refined velocity calculation. More so, the use of a less accurate velocity calculation in the dynamic programming algorithm revealed the limitations of this solution method. Because the optimization portion of the solution algorithm is so CPU-time intensive, other calculations (e.g. hydraulics) must be simplified (Brown and Koussis, 1987).

The dynamic programming approach is also limited by its procedure of choosing from discrete intervals to find a solution. Dynamic programming achieves increased accuracy by solving a problem given a large increment size and then decreasing the increment size. As discussed earlier, the new and more densely spaced alternative states span the previously found best solution and then the problem is resolved. This procedure is continued until the desired accuracy is obtained. If the increment size is decreased too rapidly, however, this method can select a local optimum rather than the desired global optimum at each stage (Labadie, 1987). The spreadsheet solution method offers no restraint from finding a similar local optimum, but neither does it prevent the trial of other alternative solutions.

The spreadsheet design method also allows use of the flexibility which is characteristic of hand design calculations. The dynamic programming algorithm chooses from alternative crown elevations of the pipe. Knowing the pipe slope and the needed flow capacity, full-flowing pipes are assumed and the pipe size is calculated with Manning's equation. This pipe size must then be rounded up to the next larger available commercial pipe size. However, a pipe size larger than the next available size may be desired in order to meet a maximum velocity constraint. Since a given design flow almost always has a lower velocity while flowing through a larger pipe (given a constant slope), the use of the larger pipe may produce a less-costly solution where the smaller pipe would violate the problem constraints. The dynamic programming algorithm would never examine the alternative of using a pipe size which is larger than required to handle the design flow. The spreadsheet design method adds this flexibility by allowing both pipe slopes and pipe sizes to be varied independently.

Despite differences in calculations and methodology, however, the spreadsheet solution has been shown to produce a less-costly solution which more consistently meets the defined problem constraints. It has been suggested that a major reason dynamic programming algorithms have seen limited use in solving real design problems has been the difficulty in defining the problem constraints. This criticism seems to be supported by this problem where the less rigorous velocity calculation limited the ability of these two algorithms to consistently meet the defined constraints of the problem. In the spreadsheet method, the velocity and cover constraints are checked manually before moving to the next upstream pipe. Automated criteria checks have been developed on the spreadsheet, but have seen limited use in favor of the quicker "eye check" method. The automated checks have been useful, however, as a last check once a design has been completed.

Another advantage of a manual constraint check is in the ability to allow a constraint to be broken. Cases may be found where a constraint is violated by a very small amount. In the spreadsheet method this exceedence may be judged acceptable (or cost effective) whereas a computerized algorithm would treat all exceedences similarly. In dynamic programming algorithms, a large penalty cost is added to the system cost when a problem constraint is broken. In this way, a solution which has violated a constraint will not be chosen unless no other feasible solution exists. In a problem without another feasible solution this methodology would favor a result which violates the fewest constraints. A "better" solution may exist, however, that violates the constraints more times but by a much smaller margin. It is difficult to program all of the judgement necessary to design a drainage system into a single algorithm.

The inherent desire of an engineer to review calculations also gives the spreadsheet method an advantage over dynamic programming algorithms. An answer obtained by a dynamic programming algorithm requires that the engineer must trust the programmer as well as the computer (Klemes, 1979). In the spreadsheet, the equation for any calculation may be checked by moving the cursor to the cell which contains the calculation. Also, since no optimization technique is employed, only a basic understanding of the hydraulic computations and the spreadsheet is needed rather than an advanced mathematics and programming background. In summary, the spreadsheet returns the control of the design procedure to the engineer.

Lastly, it is often desirable to select a non-optimal but more applicable solution in an engineering design. At times the second or tenth least-costly solution may have benefits which are not easily quantified in terms of dollars. The spreadsheet design method is not hindered by this situation. Unlike a partial enumeration optimization algorithm where the code would need to be altered substantially, the spreadsheet procedure can easily enumerate and rank alternative designs.

Despite its benefits, the spreadsheet method is limited by the number of combinations tried in a problem solution. There is no assurance that the solution found using the spreadsheet design method is the least-cost solution. With the availability of automatically updated cost estimates during the design, however, it may be assured that the spreadsheet design method will help provide a good understanding of the tradeoffs found in the drainage design problem.

Conclusions

The heuristic method used in the spreadsheet design procedure is capable of producing "optimal" designs which are better than those produced by the formal optimization algorithms. The increased flexibility of varying pipe sizes and slopes independently provides the ability to test solution alternatives which may not be tried by a more rigid partial enumeration algorithm. The spreadsheet design procedure allows the user to employ "engineering judgement" which is not easily programmed into a computer algorithm. Also, the spreadsheet design procedure emphasizes the engineering calculations rather than the algorithm needed to perform a stormwater system design. The problem definition is not lost in a solution algorithm which is difficult to understand and which must be trusted.

CHAPTER 4
USE OF ITEMIZED COST ESTIMATES AS A DESIGN HEURISITC

Importance of Cost Estimates in Design

Cost estimation is an important part of the analysis for a stormwater drainage system. With drainage costs of a highway project being estimated as up to 25% of the project costs (Linsley, 1986), the importance of drainage design becomes great. By comparison, the Florida Department of Transporation (FDOT) estimates the drainage costs of a typical highway project to be approximately 5% (FDOT, 1986). It is difficult to determine the actual percentage of a highway project cost which may be attributable to drainage. The role of items such as curb and sod are multiple. Also, because the bidding procedure allows contractors to bid on a highway project as a whole, a bid cost may not accurately reflect the true cost of the drainage. A bid on the drainage portion of the project may be low/high while the contractor makes up for the loss/gain in another portion of the project cost. A recognized increase in the relative cost of drainage in a project, however, may proportionately increase the relative importance of the engineering needed in the drainage design.

When designs are performed manually such as in the FDOT procedure, it is difficult to incorporate cost as a design criterion. In this case, the cost estimate of a project is not calculated until a feasible design for the system has been completed. Cost is included only by employing rules of thumb in the design procedure. For example, the strategy of keeping the pipe slopes as close as possible to the ground slope tends to minimize excavation costs. Similarly, the smaller the pipe diameter used, the less the cost of that pipe. These rules of thumb are helpful in designing systems that tend to be less-costly, but are poor substitutes for computing cost estimates of a system. Ideally, the engineer is provided with a cost estimate of a system while the design process is under way.

A system cost estimate that is automatically updated during the system design enables the engineer to employ a heuristic approach. A change in the design results in immediate feedback to determine whether the change is economically desirable. If a design change produces a large savings in the system cost, even if it does not immediately meet all other design constraints, it is worth redesigning other portions of the system to allow the change to be made. The ability to employ engineering judgement to a system design is an important characteristic of this algorithm.

This iterative optimization of the drainage system is performed primarily to save on costs while not decreasing the reliability. Any design change which reduces the flow capacity of a pipe in the system in some way reduces the reliability of the system. A concept which is not implemented in this spreadsheet template, but which is important in the design of drainage systems, is the consideration of risk and uncertainty in the system design. As the reliability of the system descreases, the cost of the risk or potential for system failure increases. This consideration has been used in stormwater drainage design models (Yen et al., 1976), but has had little real application by design engineers.

The spreadsheet procedure has the potential to incorporate such advanced concepts into a design format which is similar to the manual procedure presently used.

Literature Review

Cost estimating algorithms can be large mainframe based programs that involve many computations. For example, the MAPS (Methodology for Areawide Planning Studies) program is a large Fortran program which consists of several modules (Walski, 1980). Each design module produces a preliminary cost estimate for a specific type of water resource project such as a force main, gravity main, pump station, or open channel. MAPS performs planning level cost estimates using cost functions that are based on many design parameters.

Automated cost estimation of drainage designs has been incorporated into many computerized design algorithms. Some of these algorithms use functions for cost estimation. Desher and Davis (1986) use a cost function based on pipe diameter and invert depth in their heuristic design approach. The simplest cost estimate functions give cost as a function of pipe diameter (Grigg and O'Hearn, 1976; Arnell, 1982). Others include parameters such as invert depth and flow (Tyteca, 1976; Han et al., 1980).

Many of the dynamic programming algorithms use simple cost functions that require little computational effort compared to the overall solution algorithm (Zepp and Leary, 1969; Meredith, 1973; Merritt and Bogan, 1973). More recently, however, Yen et al. (1984) included the ability to enter itemized unit costs into their dynamic programming algorthm. This feature solves the problem of encoded cost functions becoming outdated.

Spreadsheet Detailed Cost Estimation

Regardless of the number of parameters used in cost functions, however, the accuracy of the cost estimate is limited. If used as a design criterion, these functions may deceive the user into believing that a parameter is of more or less importance than it actually is. This type of function fitting is convenient and easy to use and so has endured. However, now that personal computers have become so powerful in terms of storage capacity and speed of data retrieval, it is equally convenient and more accurate to perform an itemized cost estimate during the design procedure. The only way to maintain the subtle relationships found in the costs of pipes and their installation is to use the most current and accurate itemized cost data available. Design decisions should be based upon the true relationships found in the real cost figures for the items in the designed system.

The spreadsheet provides an escape from the functionalized cost estimates which may depend on one or two parameters. A detailed cost estimate of a drainage system includes quantities and unit costs of installed pipes, inlets, and manholes. The spreadsheet template updates the system cost to reflect changes in the design by using FDOT itemized average bid data. With this ability, the engineer may quickly find the design areas in the system with the largest potential savings and may easily define tradeoffs between parameter refinement and system costs.

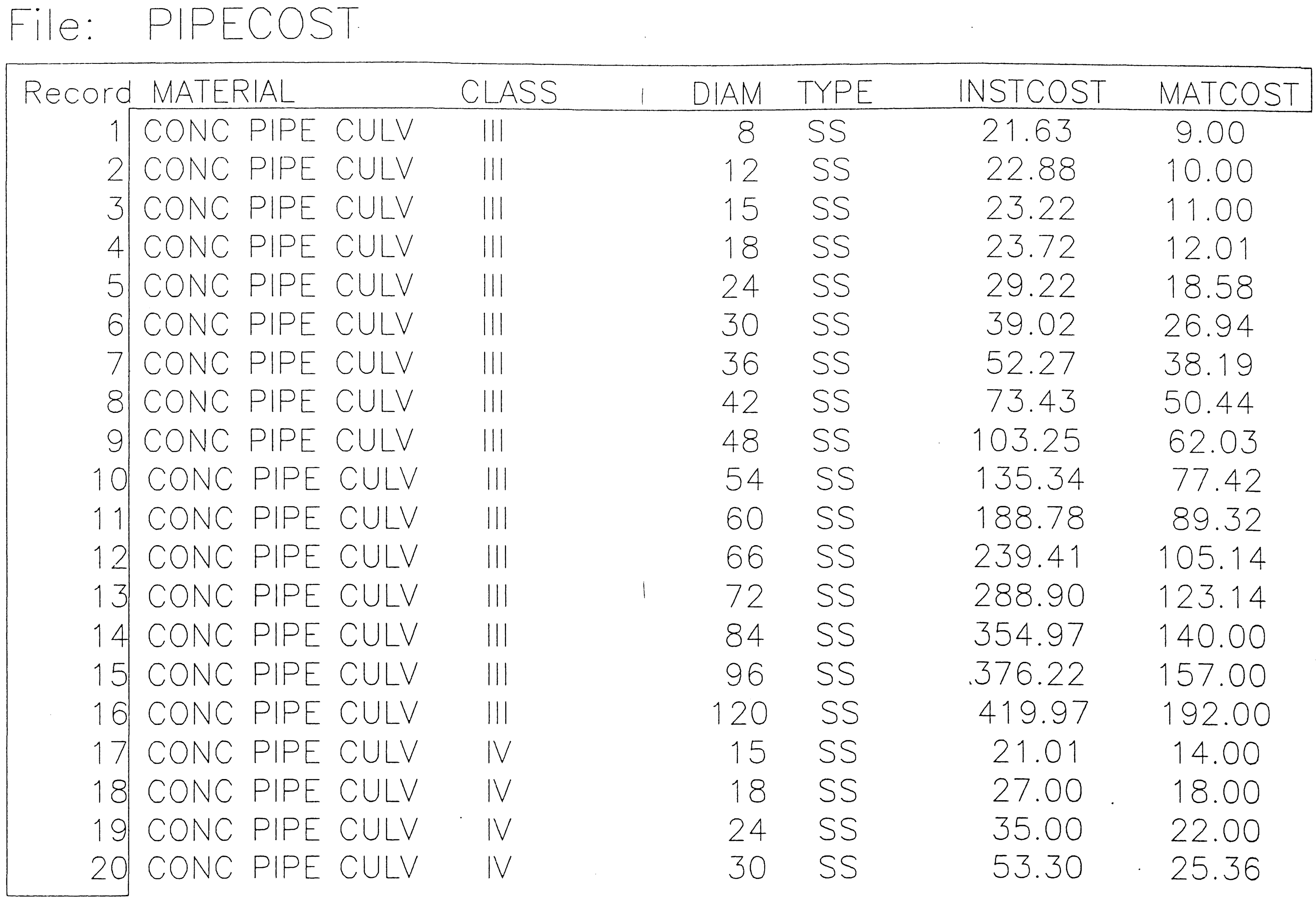
The first step in reducing costs is to acquire a feel for the distribution of cost within the system. If a detailed cost estimate is known during the design stage, then the search may concentrate on the areas of most probable savings. For example, more effort should be spent on reducing a 200 foot, 72 inch diameter pipe than on reducing a 50 foot, 18 inch diameter pipe.

Use of @Base as Cost Estimating Database

The cost estimating capability of the spreadsheet design template is performed with the use of the Lotus 1-2-3 add-in @Base (Personics Corporation, 1987). @Base is a separate software package that is loaded into the resident memory of the microcomputer and provides commands which may be accessed from within Lotus 1-2-3. The @Base program provides a command tree much like that of Lotus 1-2-3 which allows data from an external database file to be accessed from within the spreadsheet. This capability frees the spreadsheet from its inherent problem of storing very large amounts of data. @Base also provides the use of several of its own "@" functions that may access the external database without the use of the command tree.

Access to an external database adds a new dimension of power to Lotus 1-2-3. Previously, the ability to access a database from Lotus 1-2-3 meant that the entire database must be stored on every spreadsheet file from which it was accessed. The @Base feature allows the same databases to be accessed from any spreadsheet template. This ability saves greatly in storage space.

This external database capability is used by the drainage design template in the cost estimating facility. The FDOT itemized cost database, which contains average bid prices for all drainage items, is contained in two external databases. The first database contains pipe cost data (see Figure 4-1) and the second database contains structure cost data. The FDOT database contains average bid data for the material costs and installed costs for the drainage items. It does not



File: PIPECOST

Record	MATERIAL	CLASS	DIAM	TYPE	INSTCOST	MATCOST
1	CONC PIPE CULV	III	8	SS	21.63	9.00
2	CONC PIPE CULV	III	12	SS	22.88	10.00
3	CONC PIPE CULV	III	15	SS	23.22	11.00
4	CONC PIPE CULV	III	18	SS	23.72	12.01
5	CONC PIPE CULV	III	24	SS	29.22	18.58
6	CONC PIPE CULV	III	30	SS	39.02	26.94
7	CONC PIPE CULV	III	36	SS	52.27	38.19
8	CONC PIPE CULV	III	42	SS	73.43	50.44
9	CONC PIPE CULV	III	48	SS	103.25	62.03
10	CONC PIPE CULV	III	54	SS	135.34	77.42
11	CONC PIPE CULV	III	60	SS	188.78	89.32
12	CONC PIPE CULV	III	66	SS	239.41	105.14
13	CONC PIPE CULV	III	72	SS	288.90	123.14
14	CONC PIPE CULV	III	84	SS	354.97	140.00
15	CONC PIPE CULV	III	96	SS	376.22	157.00
16	CONC PIPE CULV	III	120	SS	419.97	192.00
17	CONC PIPE CULV	IV	15	SS	21.01	14.00
18	CONC PIPE CULV	IV	18	SS	27.00	18.00
19	CONC PIPE CULV	IV	24	SS	35.00	22.00
20	CONC PIPE CULV	IV	30	SS	53.30	25.36

Figure 4-1. Pipe Cost Database File Seen Using @Base Add-in Feature to Lotus 1-2-3 (Personics Corp, 1987)

explicitly provide the excavation costs, since this cost is one of several costs contained in the installation costs. Therefore, in order to be able to analyze the tradeoff between pipe and excavation costs, an estimate for excavation costs per cubic yard is obtained from the Dodge Unit Cost Data (1987). From this guide an estimate of $12.00 per cubic yard is derived by assuming that a 1/2 cubic yard backhoe is used for the excavation. This estimate also includes backfilling.

A cost estimate is performed in the spreadsheet template for each individual pipe and structure and is given directly to the right of the design calculations (see Figure 2-5). The unit cost per foot of pipe is extracted from the cost database using an @Base "@" function given the pipe diameter as the criterion. This value is multiplied by the pipe length to give the total material cost of the pipe. The required excavation needed to install the pipe is calculated by assuming that the trench width is the pipe diameter plus one foot. The total cost of the installed pipe is estimated by adding the excavation cost and the pipe cost. Since the excavation costs are not as easily calculated for a structure, the estimated fully installed cost provided in the FDOT database is used.

Recording of Previous System Designs on Spreadsheet

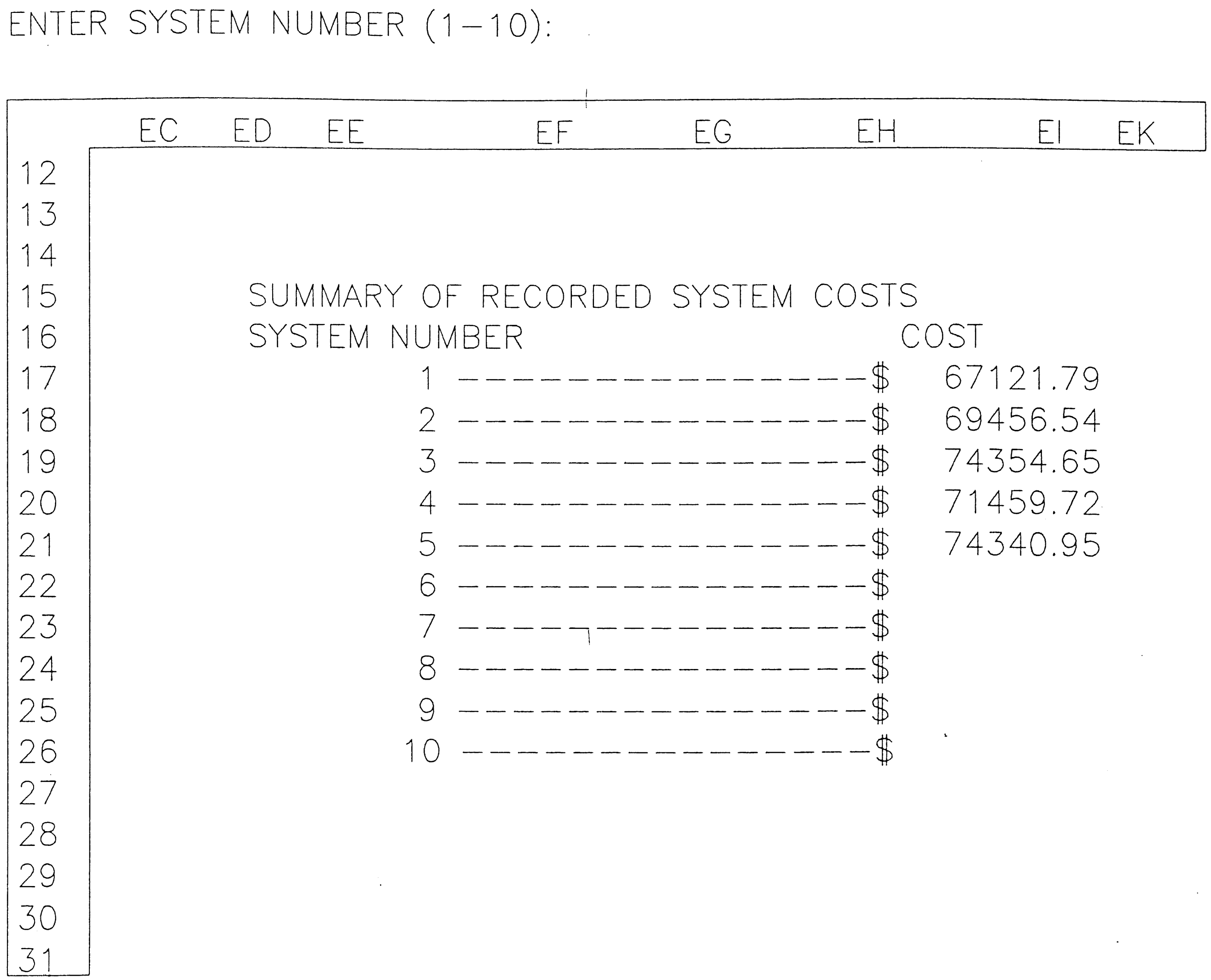
Since the heuristic design method employed by the drainage template is based on manually varying pipe parameters, assistance is provided for remembering previously tried alternatives. A save feature is provided in the template menu which records the pipe parameters and cost estimate for previous trials of a design before attempting a design change. Once the parameters of the current system are recorded, this design may be easily retrieved if a design change proves infeasible or undesirable.

At present, up to ten system designs and their cost estimates may be recorded in the template at once. A reference table (see Figure 4-2) which lists the cost estimates of the recorded designs may be addressed while retrieving a design.

The ability to save previous trials of a system design, in a sense, simulates a dynamic programming algorithm. While progressing from pipe to pipe (or node to node), the least cost solution may be saved for each alternative from the previous node. This procedure closely simulates the procedure performed by a dynamic programming algorithm, but has the advantage of further reducing the number of alternative designs attempted by applying engineering judgement and personal design strategies.

Since the spreadsheet provides direct cost feedback with a change in design, least-cost strategies are easily developed for a problem. These strategies help to reduce this trial and error approach to a much more efficient algorithm. The ability to quickly and easily calculate several solution alternatives helps to develop good engineering judgement and a good understanding for tradeoffs which exist in the problem.

If two alternatives for a pipe section are feasible (e.g. a larger pipe at a smaller slope and a smaller pipe at a steeper slope), a choice must be made based on the cost estimate for the system. Since the choice of pipe size in one section may affect the pipe size in nearby sections, design changes should be made so as to isolate the alternative systems. In other words, to truly evaluate the cost difference between the two alternatives, all resulting design changes from each alternative should be evaluated in the upstream and downstream directions until the systems again become identical. In actuality, however, this is not



```
ENTER SYSTEM NUMBER (1-10):

      EC   ED   EE          EF         EG         EH          EI   EK
12
13
14
15          SUMMARY OF RECORDED SYSTEM COSTS
16          SYSTEM NUMBER                            COST
17                     1 ------------------$   67121.79
18                     2 ------------------$   69456.54
19                     3 ------------------$   74354.65
20                     4 ------------------$   71459.72
21                     5 ------------------$   74340.95
22                     6 ------------------$
23                     7 ------------------$
24                     8 ------------------$
25                     9 ------------------$
26                    10 ------------------$
27
28
29
30
31
```

Figure 4-2. Reference Table of Saved System Costs in Drainage Design Template necessary. With the judgement acquired by performing these calculations frequently, the potential benefits and costs of a design change may be estimated for the upstream and downstream directions. Therefore the cost differences for a small number of pipes sections will give an adequate understanding with which to make a choice.

Thomasville Highway Case Study

Thomasville Highway was a reconstruction project which consisted of widening a two lane rural highway into a four lane urban highway in Tallahassee, Florida. The project was carried out by the FDOT in three phases over a span of six years.

Several characteristics of the Thomasville Highway project made it a desirable case study. The topography of the area showed significant relief for Florida (maximum of 3% grade) and so there existed the potential for a pipe size and slope tradeoff in the system design. The availability of the project blueprints and drainage planning calculations allowed an analysis of the design procedures. Since the drainage design calculations were given on the tabulation forms, a direct comparison to the spreadsheet design procedure could be made. Each of the three phases of the project consisted of approximately fifty pipes. The original design of these systems was performed manually on the tabulation forms using no explicit cost considerations during the design procedure. The blueprint versions of the drainage design reflected considerable change over the original tabulation form design. Intermediate documentation of these changes was unavailable, and it is presumed that they were also performed manually.

The FDOT Long Range Estimating (LRE) Manual approximation was applied to the Thomasville Highway Project. The typical LRE road sections are shown in Table 4-1 with their estimated drainage cost. These drainage costs are preliminary estimates and the user can apply cost factors to the drainage component according to individual project demands. The estimated drainage cost per mile is much lower than the cost estimate derived with the FDOT itemized average bid database. The itemized cost estimate is two to four times higher than the LRE estimate. A preliminary cost estimate which is very low may affect the amount of the project engineering costs that are applied to drainage. It is important to obtain as accurate of a detailed estimate as possible in the planning and design phase of a project. Drainage engineers are often caught in a bind between performing good designs and saving on costs. It is typical that engineering costs are allocated among project components in proportion to the total estimated cost of the component. Therefore, an engineer who saves over expected construction costs in a drainage design is viewed as spending too much in engineering costs. This dilemma must be overcome by the use of a more efficient design procedure. The spreadsheet design template offers this feature.

An itemized cost estimate was performed on the original FDOT design of phase 3517 of the Thomasville Highway project (see Table 4-2). This phase of the project was redesigned using the spreadsheet drainage design template with no changes being made in structure types or sizes. As seen in Table 4-2, the redesign of this phase produced a cost estimate of the pipes in the system which was 10% less than that of the original design. Both of the cost estimates were made using the FDOT itemized average bid data.

Table 4-1. FDOT Long Range Estimates of Thomasville Highway Project

(DOT) LONG RANGE ESTIMATOR

THE 8 TYPICAL ROAD TYPES ARE:

RSU = Resurfacing Undivided Median
RSD = Resurfacing Divided Median
WNU = Widening Undivided Median
WND = Widening Divided Median
NUR = New Construction Undivided Median, Rural Location
NDR = New Construction Divided Median, Rural Location
NUU = New Construction Undivided Median, Urban Location
NDU = New Construction Divided Median, Urban Location

The Basic Drainage Cost for Each Typical Follows:

RSU = $2,000 per mile
RSD = $4,000 per mile
WNU = $2,000 per mile
WND = $4,000 per mile
NUR = $60,000 per mile
NDR = $67,000 per mile
NUU = $100,000 per mile
NDU = $250,000 per mile

Thomasville Highway Drainage Cost Estimates: (NDU)

Project #	Project Length (miles)	LRE Cost Estimate	Itemized Cost Estimate	Cost/mile
3506	1.362	$340,500	$653,254	$479,629
3516	1.017	$254,250	$1,078,874	$1,060,840
3517	0.709	$177,250	$642,202	$905,785
Totals =	3.088	$772,000	$2,374,331	$768,889

Table 4-2. Comparison of Pipe Cost Estimates for Thomasville Highway Project

Itemized Cost Estimation of Thomasville Highway Project

Project No. 3517

Original Design:

Quantity	Unit	Description	Class	Size	Cost/LF	Total Cost
1196	LF	CONC PIPE CULV	III	15" SS	$23.22	$27,771
410	LF	CONC PIPE CULV	III	18" SS	$23.72	$9,725
58	LF	CONC PIPE CULV	III	24" SS	$29.22	$1,695
24	LF	CONC PIPE CULV	III	30" SS	$39.02	$936
380	LF	CONC PIPE CULV	III	36" SS	$52.27	$19,863
1640	LF	CONC PIPE CULV	III	42" SS	$73.43	$120,425
425	LF	CONC PIPE CULV	III	48" SS	$103.25	$43,881
950	LF	CONC PIPE CULV	III	54" SS	$135.34	$128,573
318	LF	CONC PIPE CULV	III	60" SS	$188.78	$60,032
217	LF	CONC PIPE CULV	III	72" SS	$288.90	$62,691
				Total Pipe Cost		$475,593

Redesign with Spreadsheet Procedure:

Quantity	Unit	Description	Class	Size	Cost/LF	Total Cost
1777	LF	CONC PIPE CULV	III	15" SS	$23.22	$41,262
33	LF	CONC PIPE CULV	III	18" SS	$23.72	$783
32	LF	CONC PIPE CULV	III	24" SS	$29.22	$935
455	LF	CONC PIPE CULV	III	30" SS	$39.02	$17,754
1048	LF	CONC PIPE CULV	III	36" SS	$52.27	$54,779
426	LF	CONC PIPE CULV	III	42" SS	$73.43	$31,281
1093	LF	CONC PIPE CULV	III	48" SS	$103.25	$112,852
535	LF	CONC PIPE CULV	III	54" SS	$135.34	$72,407
200	LF	CONC PIPE CULV	III	60" SS	$188.78	$37,756
90	LF	CONC PIPE CULV	III	66" SS	$239.41	$21,547
120	LF	CONC PIPE CULV	III	72" SS	$288.90	$34,668
				Total Pipe Cost		$426,024
				Percent Savings:		10.4%

Conclusions

It is computationally unrealistic that a manual calculation procedure such as the FDOT tabulation form would include a facility for including cost as a design criteria. A computerized algorithm is necessary to include cost estimation in the design procedure. Computerized algorithms, however, can be difficult to implement either because they are difficult to use or difficult to understand. Even when they are used, the functionalized cost estimates usually employed by these algorithms may lead the engineer to incorrect conclusions about the trade-offs found in a drainage design problem. The spreadsheet design template provides an environment where the manual procedure is closely replicated, but the advantages of the computerized algorithm are included. This environment provides an itemized cost estimate during the design which may be used as a design criterion. This cost estimate realistically reflects the trade-offs between pipe and excavation costs in the drainage design and gives a good understanding of these relationships for each individual problem. The spreadsheet procedure offers the advantages of computerization such as speed of computation without taking away from the engineer an understanding of the design algorithm.

CHAPTER 5
LINKING THE SPREADSHEET DRAINAGE DESIGN METHOD TO SWMM BY USE OF LOTUS 1-2-3 BASED PREPROCESSING

Motive Behind Linkage

The original drainage design template was developed to replicate the design procedure most commonly used by the Florida Department of Transportation. This design procedure uses the Rational Method to determine the runoff volumes and flow routing through the network. The Rational Method, though widely used, contains several limiting assumptions and is often misused (Whipple et al., 1983; Cunningham, 1987). Therefore, in order to increase the validity of the final design results from the spreadsheet design procedure, it is desirable to use a more state of the art method to simulate the runoff volumes and to perform the flow routing used in the design procedure.

The EPA Stormwater Management Model (SWMM) (Huber et al., 1981) was chosen to perform the runoff volume analysis because it is a well-established, well maintained hydrologic model which has been very popular (James et al., 1986; Miles et al., 1986). Also, its recent availability in a personal computer version from several sources has made SWMM more widely available and more easily accessible to a broader spectrum of user groups.

Table 5-1. Excerpt from SWMM User's Manual (Huber et al., 1981)

Card Group	Format	Card Columns	Description		Variable Name	Default Value
H1	8F5.0	16-20	Width of subcatchment, ft (m). This term actually refers to the physical width of overland flow in the subcatchment and may be obtained as illustrated in the text.		WW(1)*	None
		21-25	Area of subcatchment, acres (ha).		WWAREA=WW(2)*	None
		26-30	Percent imperviousness of subcatchment, %		WW(3)*	None
			Ground slope, ft/ft (dimensionless).		WSLOPE=WW(4)*	None
			Impervious area.	Roughness factor (Manning's n)	WW(5)*	None
			Pervious area.		WW(6)*	None
			Impervious area.	Depression storage, in. (mm).	WWSTORE=WW(7)*	None
			Pervious area.		WWSTORE=WW(7)*	None
	*** Horton equation parameters if INFILM = 0 (Card B1) ***					
		56-60	Maximum (initial) infiltration rate, in./hr (mm/hr).		WLMAX=WW(9)*	None
		61-65	Minimum (asymptotic) infiltration rate, in./hr (mm/hr).		WLMIN=WW(10)*	None
	F10.5	66-75	Decay rate of infiltration in Horton's equation, 1/sec.		DECAY=WW(11)*	None
	*** Green-Ampt equation parameters if INFILM = 1 (Card B1) ***					
	2F5.0	56-60	Capillary suction, inches (mm) of water.		SUCT=WW(9)*	None
		61-65	Hydraulic conductivity of soil, in./hr. (mm/hr).		HYDCON=WW(10)*	None
	F10.5	66-75	Initial moisture deficit for soil, volume air/volume voids.		SMDMAX=WW(11)*	None
H2			Blank card (except for identifier) to terminate subcatchment cards: one card.			

made. This parameter entry procedure is still in use, though it is quickly becoming obsolete.

The original parameter entry procedure has been improved upon by James and Robinson (1985) in their personal computer version of SWMM. In this version the user is prompted to enter card groups in their correct order as is shown in the example screen in Table 5-2. Prompts for the correct number of each card type are also included according to previously entered parameter values, but the user is responsible for entering the parameters in the correct order. The user's manual must be consulted to determine the ordering of the parameters on a given card. This program does, however, relieve the user of correctly spacing the entered parameters. The parameters on a card need only to be separated by commas, and default values are automatically inserted for parameters not given by the user. Although an external editor (e.g. a word processor) must be used to correct erroneously entered values, this is clearly an improvement over the original SWMM input procedure.

Structure of the Lotus 1-2-3 Preprocessor

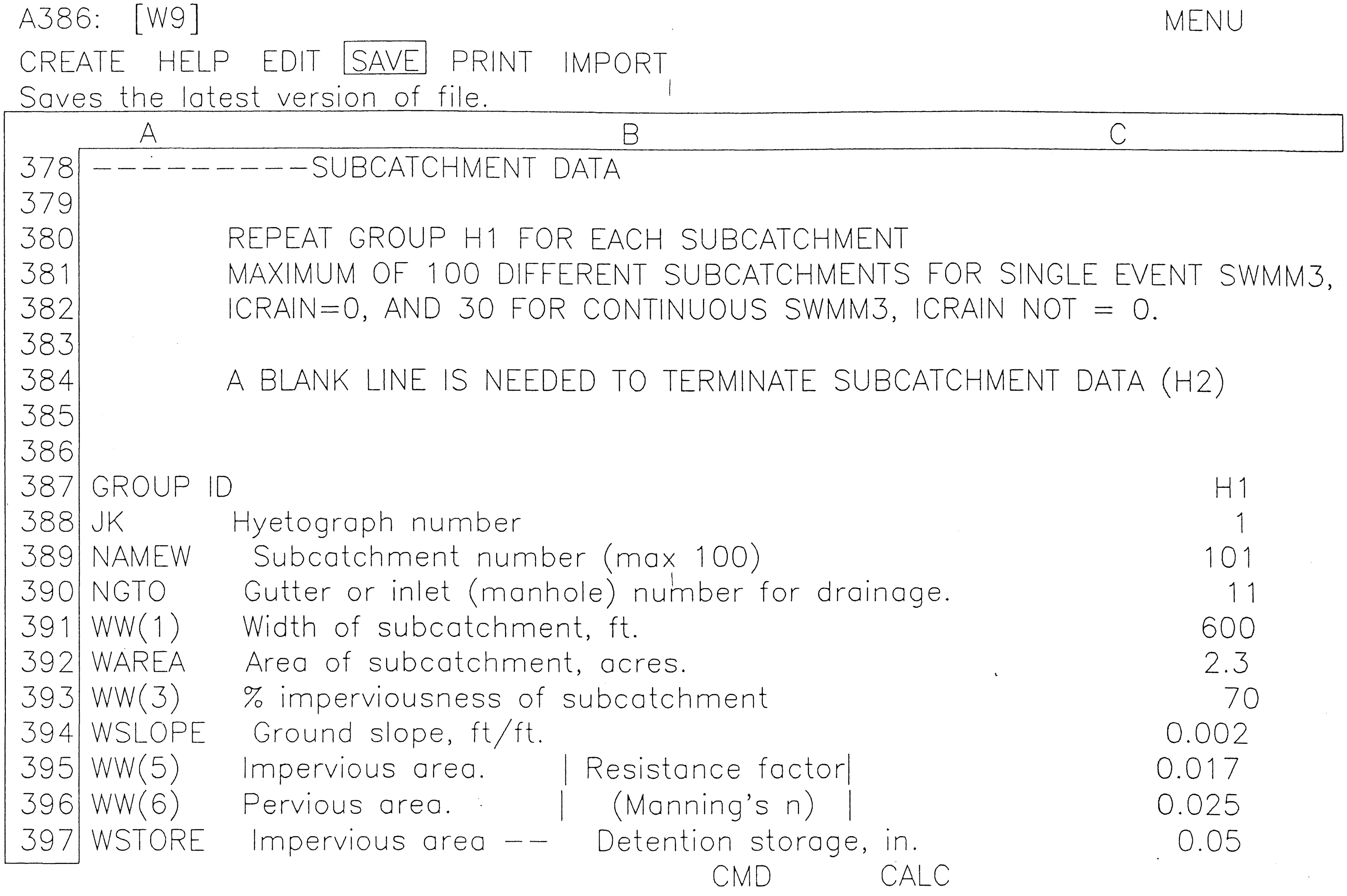
In describing their dBASE III based preprocessor for the CREAMS Model, Dennison and James (1985) present several ideas which are important in the development of a working preprocessor for a model: 1) it should be flexible and easy to use by a beginner while not hindering an expert user; 2) it should refer the user to the manual when additional information is needed; and 3) it should provide a convenient edit mode. These ideas, along with solutions to the previously described problems should be addressed in order for a preprocessor to be useful. Lotus 1-2-3 provides the flexibility needed in a software package with which these features may be conveniently implemented.

Table 5-2. Example of James and Robinson (1985) parameter entry mode.

```
  R13.  ENTER GUTTER/PIPE DATA (DATA GROUP G)             ...
G122,33,2,1.25,400,.02,0,0,.013

  R13.  ENTER GUTTER/PIPE DATA (DATA GROUP G)             ...
G133,42,2,1.25,350,.02,0,0,.013

  R13.  ENTER GUTTER/PIPE DATA (DATA GROUP G)             ...
G112,32,2,1.25,400,.013,0,0,.013

  R13.  ENTER GUTTER/PIPE DATA (DATA GROUP G)             ...
G132,42,2,1.25,430,.011,0,0,.013

  R13.  ENTER GUTTER/PIPE DATA (DATA GROUP G)             ...
G142,52,2,1.5,550,.021,0,0,.013


  TOTAL NUMBER OF GUTTERS/PIPES IS           6


  R14.  ENTER WATERSHED DATA (DATA GROUP H)               ...
H11,101,11,600,2.3,70,.002,.017,.025,.05,.2,3,.2,.0015

  R14.  ENTER WATERSHED DATA (DATA GROUP H)               ...
H11,102,22,600,2.4,70,.002,.017,.025,.05,.2,3,.2,.0015

  R14.  ENTER WATERSHED DATA (DATA GROUP H)               ...
H11,103,33,600,2.2,70,.002,.017,.025,.05,.2,3,.2,.0015
```

The preprocessor has been constructed to create an input file which is accepted by the Runoff Module of the James and Robinson (1985) version of SWMM for the personal computer (PCSWMM). In general, the preprocessor allows the user to enter parameter values in a full-screen format while the parameter name and description are in view. The preprocessor also provides an on-line help facility when the brief parameter description is inadequate. Once all parameters have been entered, then the Lotus 1-2-3 internal macro language is used to arrange the parameters into the input file format and export the file. Macro created menus are used in this template to simplify execution of the macro programs. The program menu may be called at any time while working within the preprocessor by simultaneously pressing the keys Alt-M. The following sections describe the choices presented by the menu.

Create. The "Create" mode allows the user to begin a new job file. The first page of this selection asks for input such as job initials and a job number which are used to name the formatted file created by the preprocessor. Items such as the number of subcatchments and number of gutters and pipes in the simulation are also requested in order to allocate space within the worksheet. The user is next moved to the parameter entry section of the preprocessor. As can be seen in the sample of this section shown in Figure 5-1, the value is entered just to the right of a brief description of the parameter. The SWMM parameter name is also given at the far left of the screen. In cases where more than one of a given card group is needed, such as with subcatchment cards, multiple card values are entered directly to the right of the initial card values.



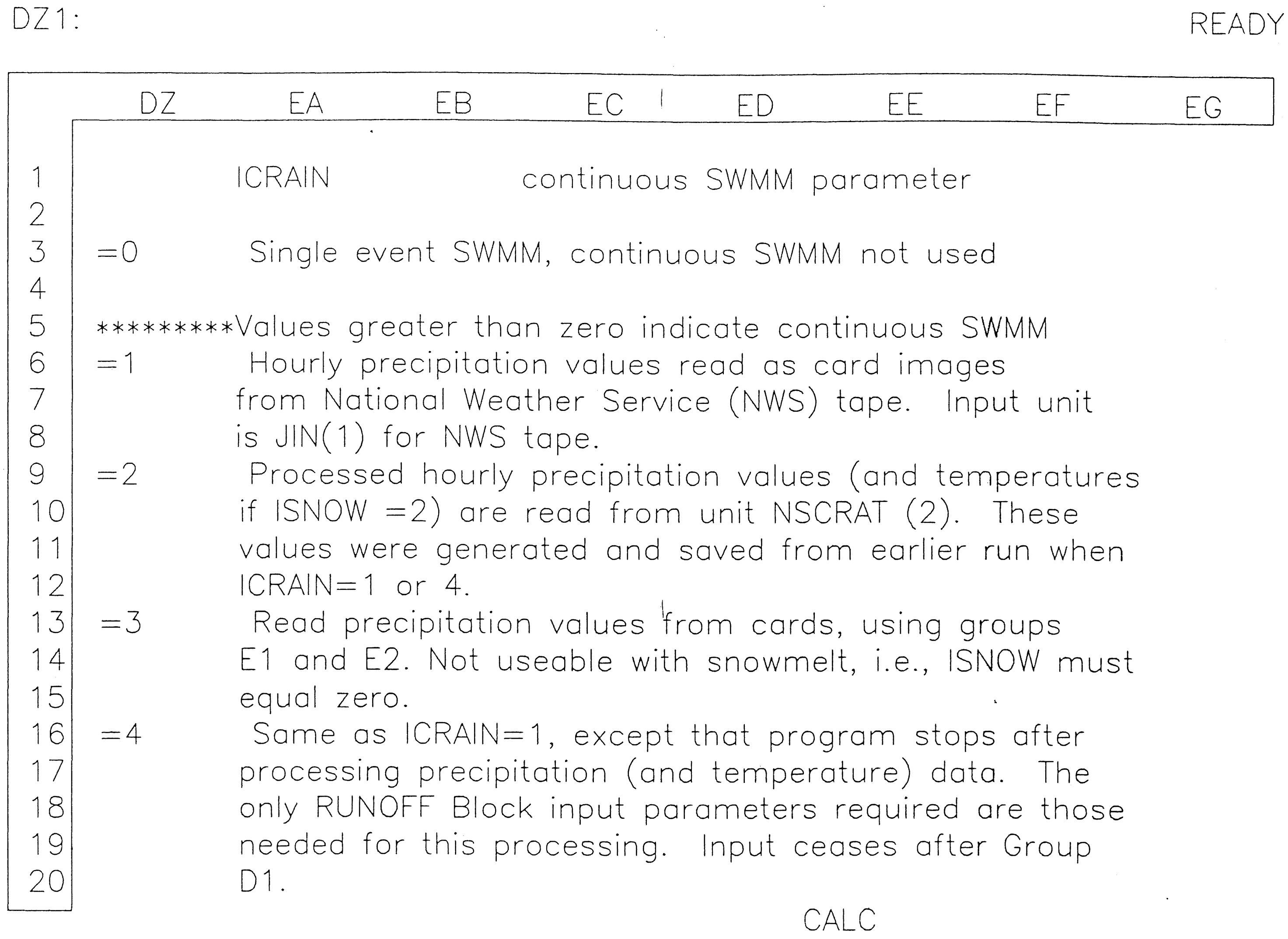
```
A386:  [W9]                                                      MENU
CREATE  HELP  EDIT  SAVE  PRINT  IMPORT
Saves the latest version of file.
          A                       B                              C
378 ----------SUBCATCHMENT DATA
379
380          REPEAT GROUP H1 FOR EACH SUBCATCHMENT
381          MAXIMUM OF 100 DIFFERENT SUBCATCHMENTS FOR SINGLE EVENT SWMM3,
382          ICRAIN=0, AND 30 FOR CONTINUOUS SWMM3, ICRAIN NOT = 0.
383
384          A BLANK LINE IS NEEDED TO TERMINATE SUBCATCHMENT DATA (H2)
385
386
387 GROUP ID                                                           H1
388 JK        Hyetograph number                                         1
389 NAMEW     Subcatchment number (max 100)                           101
390 NGTO      Gutter or inlet (manhole) number for drainage.           11
391 WW(1)     Width of subcatchment, ft.                              600
392 WAREA     Area of subcatchment, acres.                            2.3
393 WW(3)     % imperviousness of subcatchment                         70
394 WSLOPE    Ground slope, ft/ft.                                  0.002
395 WW(5)     Impervious area.     | Resistance factor|             0.017
396 WW(6)     Pervious area.       |  (Manning's n)   |             0.025
397 WSTORE    Impervious area --    Detention storage, in.           0.05
                                          CMD       CALC
```

Figure 5-1. Parameter Entry Area of Spreadsheet Preprocessor

Help. From the main menu choices, the "Help" selection allows the user to access more specific information about a given parameter. The help file for a parameter may include further documentation, calculation aids, graphs, or any other material with which the user may make a logical and defensible parameter estimate for a specific job location. Most documentation files come directly from the User's Manual (Huber et al., 1981; Roesner et al., 1983) as in the screen shown in Figure 5-2. Once the user is finished viewing any specific help file, pressing return will move the cursor back to the parameter entry mode.

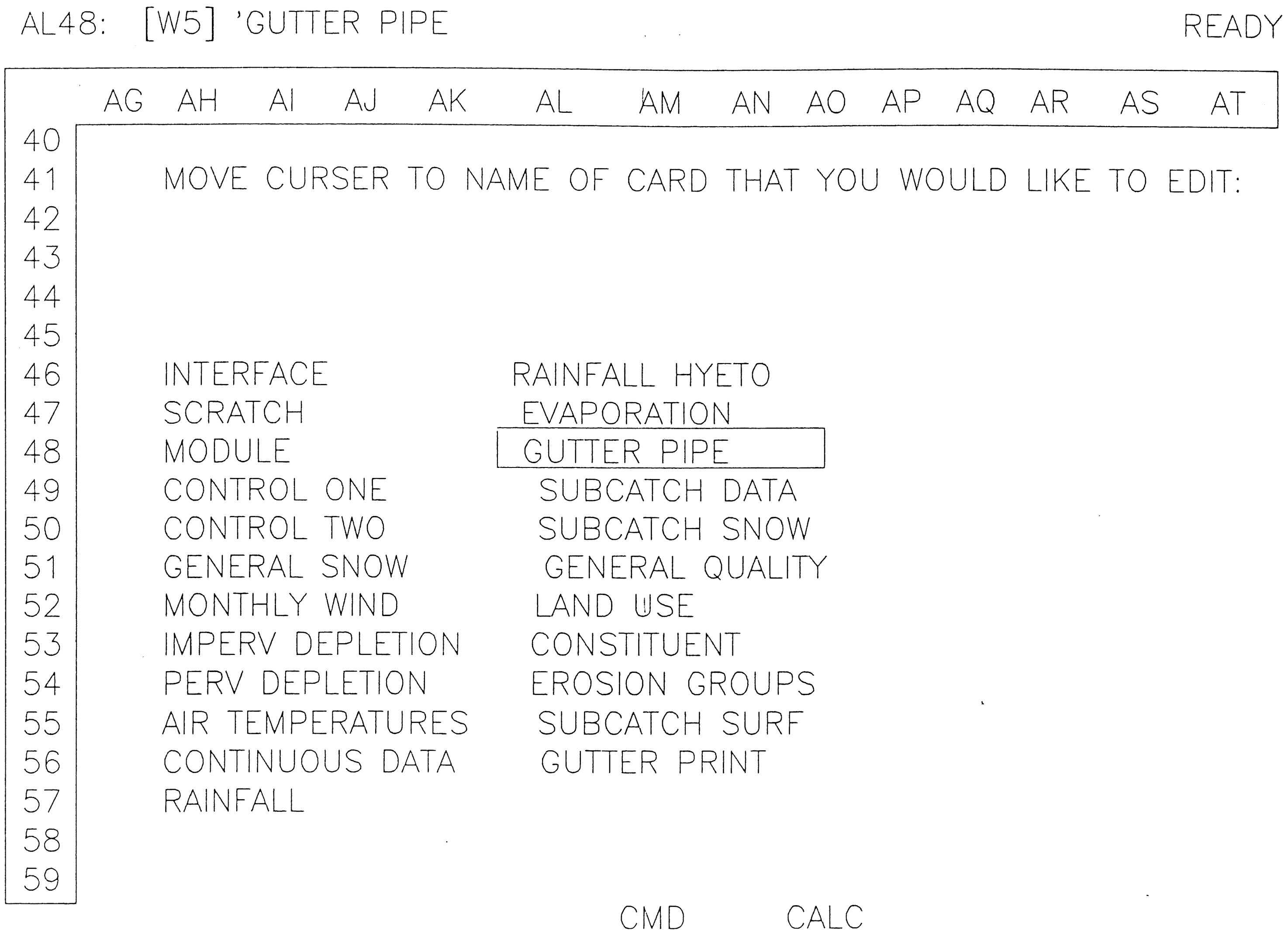
Edit. The "Edit" command is a short-cut procedure for finding the parameter entry position for a given input card type. This macro allows the user to quickly update a parameter value on a previously created file, or to return to a previous card on the present file and make a change. The user is able to move directly to any card in the module by moving the cursor to the card name on the edit screen (see Figure 5-3).

Save. The "Save" procedure has been improved considerably over the previous edition of the Runoff Module preprocessor. The time-consuming procedure of printing cards individually to files and importing them back in the worksheet has been eliminated. This procedure has been replaced by the use of Lotus 1-2-3 string arithmetic. The Lotus 1-2-3 "@string" function is used to convert the entered numeric value into a string and then string arithmetic procedures are used to combine entries into their correct format. In order to ensure that the correct number of spaces is allocated to each parameter on a line, the "@right" function is used. The @right function truncates unneeded spaces to the left of the parameter value. This step is important to preserve the correct spacing in each line of the formatted output file. Lastly, the "Save"



```
DZ1:                                                              READY

       DZ        EA        EB        EC        ED        EE        EF        EG
 1               ICRAIN                   continuous SWMM parameter
 2
 3    =0          Single event SWMM, continuous SWMM not used
 4
 5    *********Values greater than zero indicate continuous SWMM
 6    =1          Hourly precipitation values read as card images
 7               from National Weather Service (NWS) tape.  Input unit
 8               is JIN(1) for NWS tape.
 9    =2          Processed hourly precipitation values (and temperatures
10               if ISNOW =2) are read from unit NSCRAT (2).  These
11               values were generated and saved from earlier run when
12               ICRAIN=1 or 4.
13    =3          Read precipitation values from cards, using groups
14               E1 and E2. Not useable with snowmelt, i.e., ISNOW must
15               equal zero.
16    =4          Same as ICRAIN=1, except that program stops after
17               processing precipitation (and temperature) data.  The
18               only RUNOFF Block input parameters required are those
19               needed for this processing.  Input ceases after Group
20               D1.
                                                     CALC
```

Figure 5-2. Example of SWMM Help File from User's Manual (Huber et al., 1981)



```
AL48:  [W5] 'GUTTER PIPE                                                  READY

      AG   AH    AI    AJ    AK     AL     AM    AN   AO   AP   AQ   AR    AS    AT
 40
 41       MOVE CURSER TO NAME OF CARD THAT YOU WOULD LIKE TO EDIT:
 42
 43
 44
 45
 46       INTERFACE               RAINFALL HYETO
 47       SCRATCH                 EVAPORATION
 48       MODULE                  GUTTER PIPE
 49       CONTROL ONE               SUBCATCH DATA
 50       CONTROL TWO               SUBCATCH SNOW
 51       GENERAL SNOW              GENERAL QUALITY
 52       MONTHLY WIND             LAND USE
 53       IMPERV DEPLETION         CONSTITUENT
 54       PERV DEPLETION           EROSION GROUPS
 55       AIR TEMPERATURES          SUBCATCH SURF
 56       CONTINUOUS DATA           GUTTER PRINT
 57       RAINFALL
 58
 59
                                  CMD       CALC
```

Figure 5-3. Illustration of Edit Mode Menu command arranges the input cards into the order to be exported. This procedure is performed by a macro program that successively copies each card type below the previous cards in the correct order. The difficulty present in this procedure is the possibility of multiple cards of a card type (e.g. subcatchment, gutter cards). To allow for this variability, a named range is created in the spreadsheet which is sized according to an answer provided by the user on the opening screen of the preprocessor. When copying the cards to the final ordered format, the {end}{down} command is used in the macro program to ensure that the cursor is at the bottom of the previously copied range before copying the next range of cards.

<u>Print</u>. Once the formatted listing of cards has been reviewed on the spreadsheet, the "Print" command is invoked to print the formatted cards into an ASCII file which may be accepted by a personal computer version of the Runoff Module as input (see Figure 5-4). As mentioned earlier, the input files are named using the job initials and job number entered at the beginning of the "Create" mode in order to be consistent with PCSWMM (James and Robinson, 1985), which was used here to execute the SWMM code.

<u>Import</u>. This command is used when linking the preprocessor to the drainage design template. The pipe data which are stored in an external database are automatically placed in the gutter/pipe parameter card entry area. This procedure is described in more detail in the following section.

```
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
   1   2   0   0   0
RUNOFF
A1TEST SFWMD 24-HR DISTRIBUTION
A2ITERATIVE DESIGN PROCEDURE -- 20 PIPE SYSTEM
B1  0 0  0    1    0    0    0   12    0   29    2   88    0    0    0
B2     100   15   30    0    0    0
E1      97   15
E2   0.0000.0120.0180.0180.0120.0180.0120.0180.0120.018
E2   0.0180.0180.0180.0180.0180.0240.0180.0240.0240.030
E2   0.0240.0300.0300.0360.0300.0360.0360.0420.0360.042
E2   0.0420.0480.0420.0480.0480.0540.0540.0600.0600.066
E2   0.0660.0720.0720.0960.0960.1500.1501.0141.0080.216
E2   0.2220.1140.1140.0840.0840.0660.0720.0540.0480.048
E2   0.0420.0480.0420.0480.0420.0300.0240.0300.0240.030
E2   0.0240.0240.0300.0300.0240.0240.0300.0300.0240.024
E2   0.0300.0180.0180.0180.0180.0180.0180.0180.0180.018
E2   0.0180.0180.0180.0180.0180.0180.0180.0000.0000.000
G1      11   22    3   0.00       0  0.000   0.000   0.000   0.000      0
G1      22   33    3   0.00       0  0.000   0.000   0.000   0.000      0
G1      33   42    3   0.00       0  0.000   0.000   0.000   0.000      0
G1      12   32    3   0.00       0  0.000   0.000   0.000   0.000      0
G1      32   42    3   0.00       0  0.000   0.000   0.000   0.000      0
G1      42   52    3   0.00       0  0.000   0.000   0.000   0.000      0
G1      23   34    3   0.00       0  0.000   0.000   0.000   0.000      0
G1      34   43    3   0.00       0  0.000   0.000   0.000   0.000      0
G1      43   52    3   0.00       0  0.000   0.000   0.000   0.000      0
G1      52   61    3   0.00       0  0.000   0.000   0.000   0.000      0
G1      31   41    3   0.00       0  0.000   0.000   0.000   0.000      0
G1      41   51    3   0.00       0  0.000   0.000   0.000   0.000      0
G1      51   61    3   0.00       0  0.000   0.000   0.000   0.000      0
G1      61   71    3   0.00       0  0.000   0.000   0.000   0.000      0
G1      44   53    3   0.00       0  0.000   0.000   0.000   0.000      0
G1      53   62    3   0.00       0  0.000   0.000   0.000   0.000      0
G1      62   71    3   0.00       0  0.000   0.000   0.000   0.000      0
G1      71   81    3   0.00       0  0.000   0.000   0.000   0.000      0
G1      81   91    3   0.00       0  0.000   0.000   0.000   0.000      0
G1      91   10    3   0.00       0  0.000   0.000   0.000   0.000      0
G2       0    0    0   0.00       0  0.000   0.000   0.000   0.000      0
H1  1  101   11  600   2.3 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  102   22  600   2.4 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  103   33  600   2.2 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  104   12  330   1.5 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  105   32  650   2.2 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  106   42 2600   9.1 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  107   23 3200  10.2 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  108   34 2000   5.7 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  109   43 4000  11.9 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  110   31 4600  11.1 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  111   41 2200   5.5 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  112   51 1600   4.3 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  113   52 5600  15.4 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  114   61 1600   3.2 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  115   44 2800   8.0 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  116   53 1600   2.9 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  117   62 2600   7.8 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  118   71 1600   4.0 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  119   81 2400   5.5 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  120   91 2400   5.0 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H2  0    0    0    0   0.0  0.00.0000.0000.0000.0000.0000.00 0.00 0.000000
M1 20    1     0     0     0     0     0     0     0     0     0     0
M2 11   22   33   12   32   42   23   34   43   31   41   51   52   61   44   53
M2 62   71   81   91    0    0    0    0    0    0    0    0    0    0    0    0
ENDPROGRAM
```

Figure 5-4. Example Input File for Runoff Module Using SWMM "Dummy" Gutter Option

Iterative Design Procedure Based on Linkage of Drainage Design Template and SWMM Preprocessor

Mechanics of Linkage

In order to use both the drainage design template and the Runoff Module preprocessor on the same design problem, a link between the two spreadsheet templates was established. The link was performed using the @Base Lotus 1-2-3 add-in package (Personics Corp., 1987) previously described in Chapter 4. A database is defined which consists of fields describing the to and from nodes for a pipe as well as pertinent design specifications such as the pipe size, length, slope, and Manning's coefficient. Also, since this database is shared by all projects, a field was needed to define the job initials and job number.

In this procedure the data must be translated from the drainage design template to the database and then to the preprocessor. Therefore the facilities must be made in the design template for exportation and in the preprocessor for importation. Each of these processes is performed with the aid of the @Base commands. The export procedure uses the Data Transfer command provided in the @Base menu. An export area which contains a compact summary of all data to be exported has been constructed within the drainage template. Once this range is defined as the export range, then the drainage system data is appended onto the existing database.

The importation of the drainage system parameters into the preprocessor is performed using two of the "@" functions provided by the @Base add-in package. The general formats of the two functions are as follows: @dbfirst(ALIAS,CRITERIA),

@dbfld(ALIAS,FIELDNAME,RECORD NUMBER).

The "ALIAS" is the name given to the database file. The @dbfirst function returns the record number of the first record which matches the criteria. The @dbfld function returns the value in the given file which matches the given field name and record number. For the drainage system application, these two functions are combined in the following format:

`@dbfld(ALIAS,FIELDNAME,@dbfirst(ALIAS,CRITERIA)).`

With this combination the value in the given field of the first record to match the criteria is returned. For example, the following statement may be used to retrieve the length of the pipe from node "11" for the job SWM-1 (the database used in this example is named "DRAINAGE"):

```
@dbfld("DRAINAGE","LENGTH",@dbfirst("DRAINAGE",CRITERIA))
where CRITERIA = "FROM=11.AND.JOBINIT=SWM.AND.JOBNO=1"
```

These statements have already been written in the preprocessor template for each parameter and need only to be copied into the input area for the gutter data. This procedure is performed automatically by the "Import" command in the preprocesser menu. The user must then enter the criteria values to be met (e.g. enter "11" next to the cell labeled "FROM") and recalculate the spreadsheet. Other than this procedure, the preprocessor is used as before.

Steps in Iterative Design Procedure

This design procedure uses SWMM to provide initial estimates of the peak flows in each pipe of the system. These initial estimates are acquired by using the "dummy" gutter assumption in SWMM. This assumption effectively neglects any routing procedure by assuming the lengths of all pipes to be zero and by assuming the inflow of a pipe equals its outflow. The result of this assumption is a simple addition of all inflow hydrographs to a node at each time step in the simulation.

The initial estimates of peak flow in each pipe are next used to design the drainage system in the design template. Once this design is complete, the pipe design parameters (e.g. diameter, slope) are exported from the design template to the database and then imported into the SWMM preprocessor. The preprocessor is able to reproduce the previously used input file with the exception that the "dummy" gutters are replaced with the newly designed pipes. The SWMM Runoff Module may now be run with the flows being routed through the network in a more sophisticated manner. Although the Runoff Module has a less sophisticated routing procedure (non-linear reservoir) than the Transport (kinematic wave) or Extran (complete St. Venant equations) Modules (Huber et al., 1981), its low computational expense is desirable since multiple runs of the Module may be necessary in the design procedure. The Runoff Module allows for the input of rainfall hyetographs rather than simple rainfall intensities. Also, even this relatively simple routing scheme, when used with proper parameter selection and verification techniques, provides a large improvement over the currently used Rational Method. The output of the second SWMM Runoff Module simulation is then compared to the output of the first run. If the peak flow values in each pipe are similar, then the design procedure is complete. Routing the hydrographs through the newly designed pipe network will most likely, however, produce either some change in the peak flow values or a surcharge indication in a given pipe. If the system produces no surcharges in any pipes, then the new peak flows are used to redesign the drainage system. A surcharge message indicates that a pipe must be redesigned to either handle a larger flow or provide some retention or detention. The drainage design template does not, however, address the design of retention or detention facilities. From the drainage design template, any changes from a redesign are again exported through the database and into the preprocessor to be used in a new Runoff Module simulation. This iterative procedure is then repeated until the peak flows converge to a true value for the design. In this manner, the Runoff Module not only acts as a method for determining runoff hydrographs, but also as a verification of the drainage system design.

Example of Iterative Design Procedure

This section provides a more detailed step by step explanation for the iterative design procedure (see Figure 5-5) while presenting the results of an example problem. The pipe linkages, pipe lengths and ground elevations at nodes are the same as for the example problem presented in Chapter 3.

The first step in this design procedure is to define the pipe linkages by entering the "to" and "from" nodes into the drainage design template. At this time the job initials and job number are also defined. This information is next exported to the @Base database file. The next several steps involve the use of the Runoff Module preprocessor. Along with the pipe linkage information which is imported from the database, the subcatchment characteristics and the rainfall hyetograph data must be input into the preprocessor. Included in the subcatchment characteristics is the inlet number to which the runoff is routed and the subcatchment width and area. This information has been fabricated for the example problem presented and is shown in Table 5-3. These values are not critical to demonstrate the technique involved in performing this design procedure; however, realistic values for these

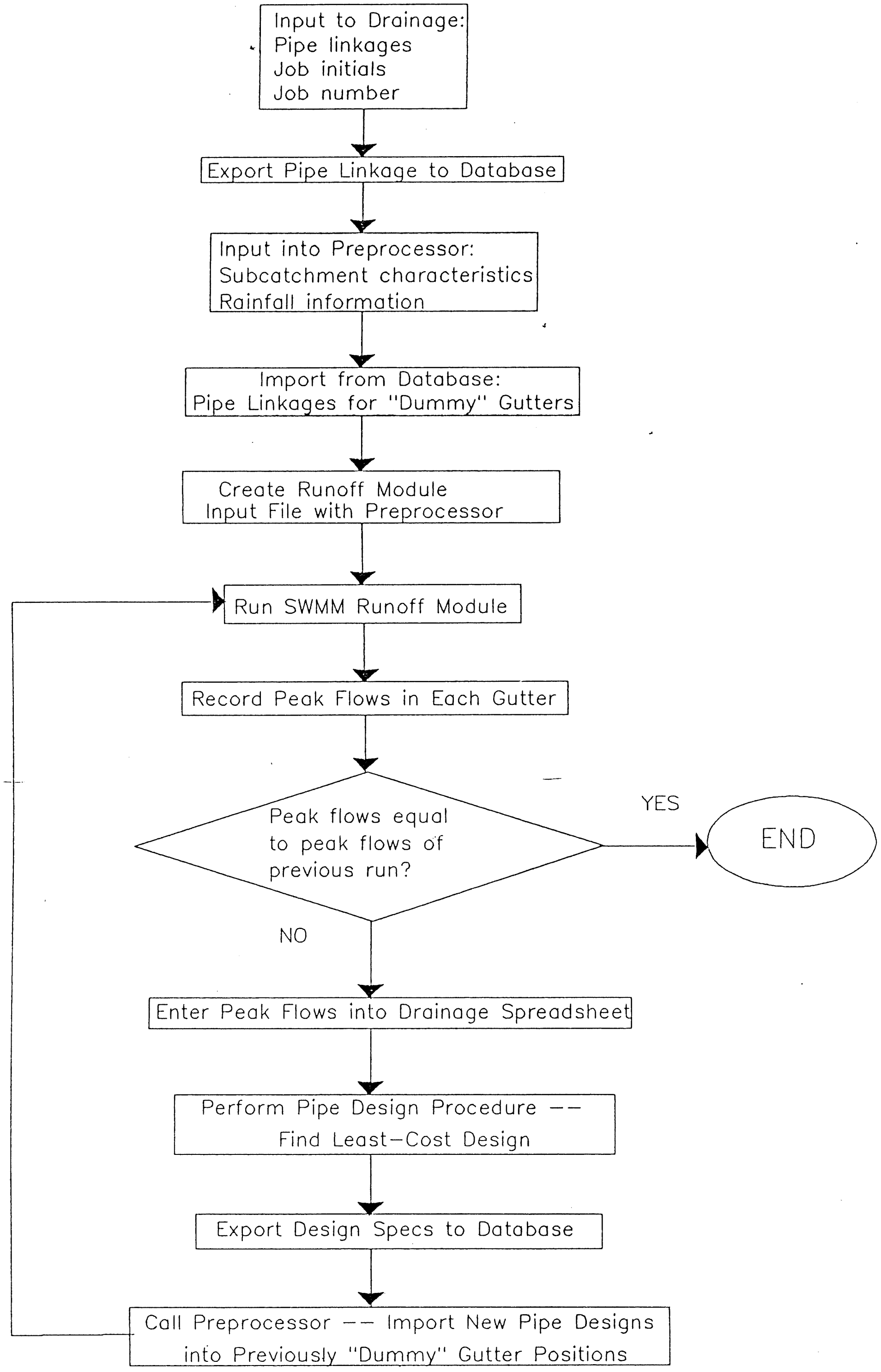


Figure 5-5. Flow Chart of Iterative Design Process

Table 5-3. Characteristics of Subcatchments Used in Runoff Module Simulation

						Manning's n ***** for	********	Detention Storage ***** for	********	Horton's Equation ************************		
Subcatchment Number	Drains to	Width (ft)	Area (acres)	Percent Impervious	Average Slope (ft/ft)	Imperv. Area	Pervious Area	Imperv. Area (in)	Pervious Area (in)	Max Infilt (in/hr)	Min Infilt (in/hr)	Decay Rate (1/sec)
101	11	600	2.3	70	0.002	0.017	0.025	0.05	0.2	3	0.3	0.0015
102	22	600	2.4	70	0.002	0.017	0.025	0.05	0.2	3	0.3	0.0015
103	33	600	2.2	70	0.002	0.017	0.025	0.05	0.2	3	0.3	0.0015
104	12	330	1.5	70	0.002	0.017	0.025	0.05	0.2	3	0.3	0.0015
105	32	650	2.2	70	0.002	0.017	0.025	0.05	0.2	3	0.3	0.0015
106	42	2600	9.1	70	0.002	0.017	0.025	0.05	0.2	3	0.3	0.0015
107	23	3200	10.2	70	0.002	0.017	0.025	0.05	0.2	3	0.3	0.0015
108	34	2000	5.7	70	0.002	0.017	0.025	0.05	0.2	3	0.3	0.0015
109	43	4000	11.9	70	0.002	0.017	0.025	0.05	0.2	3	0.3	0.0015
110	31	4600	11.1	70	0.002	0.017	0.025	0.05	0.2	3	0.3	0.0015
111	41	2200	5.5	70	0.002	0.017	0.025	0.05	0.2	3	0.3	0.0015
112	51	1600	4.3	70	0.002	0.017	0.025	0.05	0.2	3	0.3	0.0015
113	52	5600	15.4	70	0.002	0.017	0.025	0.05	0.2	3	0.3	0.0015
114	61	1600	3.2	70	0.002	0.017	0.025	0.05	0.2	3	0.3	0.0015
115	44	2800	8.0	70	0.002	0.017	0.025	0.05	0.2	3	0.3	0.0015
116	53	1600	2.9	70	0.002	0.017	0.025	0.05	0.2	3	0.3	0.0015
117	62	2600	7.8	70	0.002	0.017	0.025	0.05	0.2	3	0.3	0.0015
118	71	1600	4.0	70	0.002	0.017	0.025	0.05	0.2	3	0.3	0.0015
119	81	2400	5.5	70	0.002	0.017	0.025	0.05	0.2	3	0.3	0.0015
120	91	2400	5.0	70	0.002	0.017	0.025	0.05	0.2	3	0.3	0.0015

parameters were chosen. The rainfall distribution used for the example problem was the synthetic distribution used by the South Florida Water Management District (SFWMD) for their 24-hour storm (SFWMD, 1987). A listing of the rainfall distribution used in this simulation is shown in Table 5-4.

At this point in the design process only the linkages ("to" and "from" nodes) are known for each pipe. Therefore, preliminary peak flow estimates are acquired from the SWMM Runoff Module by using the "dummy" gutter option. The resulting peak flows in each pipe from this simulation (see Table 5-5) are used as preliminary values in the drainage design template. With these peak flows and other necessary system specifications (pipe lengths, Manning's n, ground elevation at each node), the iterative drainage system design method may be employed.

The results from the drainage design method are the sizes and slopes of a pipe network deemed capable of carrying the peak flows determined by the SWMM Runoff Module. However, if the runoff hydrographs from the subcatchments were routed through the designed pipe system rather than the "dummy" system, the calculated peak flows should differ. Therefore, the next step is to enter the newly designed pipe system into the SWMM preprocessor in place of the previously used "dummy" gutters. This step is accomplished by exporting the pipe system design specifications to the @Base database file and then importing this information into the preprocessor using the previously described "Import" command. The preprocessor now generates an input file in which the "G1" or gutter cards contain the pipe network parameters determined by the design template (see Figure 5-6). This input file may be

Table 5-4. Rainfall distribution Used in Runoff Module Simulation.

Time (hours)	intensity (in/hr)	Time (hours)	intensity (in/hr)	Time (hours)	intensity (in/hr)	Time (hours)	intensity (in/hr)
0.25	0.012	6.25	0.036	12.25	0.216	18.25	0.03
0.5	0.018	6.5	0.036	12.5	0.222	18.5	0.024
0.75	0.018	6.75	0.042	12.75	0.114	18.75	0.024
1	0.012	7	0.036	13	0.114	19	0.03
1.25	0.018	7.25	0.042	13.25	0.084	19.25	0.03
1.5	0.012	7.5	0.042	13.5	0.084	19.5	0.024
1.75	0.018	7.75	0.048	13.75	0.066	19.75	0.024
2	0.012	8	0.042	14	0.072	20	0.03
2.25	0.018	8.25	0.048	14.25	0.054	20.25	0.018
2.5	0.018	8.5	0.048	14.5	0.048	20.5	0.018
2.75	0.018	8.75	0.054	14.75	0.048	20.75	0.018
3	0.018	9	0.054	15	0.042	21	0.018
3.25	0.018	9.25	0.06	15.25	0.048	21.25	0.018
3.5	0.018	9.5	0.06	15.5	0.042	21.5	0.018
3.75	0.024	9.75	0.066	15.75	0.048	21.75	0.018
4	0.018	10	0.066	16	0.042	22	0.018
4.25	0.024	10.25	0.072	16.25	0.03	22.25	0.018
4.5	0.024	10.5	0.072	16.5	0.024	22.5	0.018
4.75	0.03	10.75	0.096	16.75	0.03	22.75	0.018
5	0.024	11	0.096	17	0.024	23	0.018
5.25	0.03	11.25	0.15	17.25	0.03	23.25	0.018
5.5	0.03	11.5	0.15	17.5	0.024	23.5	0.018
5.75	0.036	11.75	1.014	17.75	0.024	23.75	0.018
6	0.03	12	1.008	18	0.03	24	0.018

Table 5-5. Summary of Iterative Solutions to Drainage Design Problem

=========> Direction of Convergence Procedure =========>
All Pipe Flows Calculated Using Manning's Equation with n = 0.013.
All Pipe Diameters are Given in Inches.

Defined Pipe Linkages		Peak Flows from SWMM with "Dummy" Gutter Routing (cfs)	Results of First Design		Peak Flows from SWMM Routing with 1st Design (cfs)	Results of Second Design		Peak Flows from SWMM Routing with 2nd Design (cfs)
From	To		Slope	Diameter		Slope	Diameter	
11	22	1.83	0.014	15	1.83	0.015	15	1.83
22	33	3.72	0.020	15	3.54	0.020	15	3.54
33	42	5.50	0.022	15	5.37	0.020	15	5.37
12	32	1.13	0.013	15	1.13	0.013	15	1.13
32	42	2.96	0.012	15	2.8	0.011	15	2.8
42	52	15.94	0.018	24	15.13	0.021	18	15.14
23	34	8.65	0.010	18	8.65	0.010	18	8.65
34	43	13.66	0.022	18	12.63	0.022	18	12.63
43	52	23.97	0.014	24	22	0.014	24	22
52	61	53.59	0.011	36	50.43	0.010	36	49.21
31	41	10.26	0.020	18	10.26	0.020	18	10.26
41	51	15.28	0.015	24	14.34	0.019	18	14.34
51	61	19.13	0.015	24	17.24	0.014	24	17.28
61	71	75.84	0.018	36	72.34	0.018	36	71.56
44	53	7.02	0.010	18	7.02	0.010	18	7.02
53	62	9.92	0.016	18	9.21	0.014	18	9.21
62	71	16.67	0.017	24	15.38	0.022	18	15.35
71	81	96.16	0.010	42	93.02	0.011	42	92.48
81	91	101.32	0.011	42	99.31	0.011	42	98.92
91	10	106.14	0.012	42	105.22	0.012	42	105.05

```
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
   1   2   0   0   0
RUNOFF
A1TEST SFWMD 24-HR DISTRIBUTION
A2ITERATIVE DESIGN PROCEDURE -- 20 PIPE SYSTEM
B1  0 0  0    1    0    0    0   12    0   29    2   88    0    0    0
B2      100   15   30    0    0    0
E1       97   15
E2   0.0000.0120.0180.0180.0120.0180.0120.0180.0120.018
E2   0.0180.0180.0180.0180.0180.0240.0180.0240.0240.030
E2   0.0240.0300.0300.0360.0300.0360.0360.0420.0360.042
E2   0.0420.0480.0420.0480.0480.0540.0540.0600.0600.066
E2   0.0660.0720.0720.0960.0960.1500.1501.0141.0080.216
E2   0.2220.1140.1140.0840.0840.0660.0720.0540.0480.048
E2   0.0420.0480.0420.0480.0420.0300.0240.0300.0240.030
E2   0.0240.0240.0300.0300.0240.0240.0300.0300.0240.024
E2   0.0300.0180.0180.0180.0180.0180.0180.0180.0180.018
E2   0.0180.0180.0180.0180.0180.0180.0180.0000.0000.000
G1       11   22    2    1.25      350   0.015   0.000   0.000   0.013       0
G1       22   33    2    1.25      400   0.020   0.000   0.000   0.013       0
G1       33   42    2    1.25      350   0.020   0.000   0.000   0.013       0
G1       12   32    2    1.25      400   0.013   0.000   0.000   0.013       0
G1       32   42    2    1.25      430   0.011   0.000   0.000   0.013       0
G1       42   52    2    1.50      550   0.021   0.000   0.000   0.013       0
G1       23   34    2    1.50      500   0.010   0.000   0.000   0.013       0
G1       34   43    2    1.50      450   0.022   0.000   0.000   0.013       0
G1       43   52    2    2.00      350   0.014   0.000   0.000   0.013       0
G1       52   61    2    3.00      500   0.010   0.000   0.000   0.013       0
G1       31   41    2    1.50      500   0.020   0.000   0.000   0.013       0
G1       41   51    2    1.50      350   0.019   0.000   0.000   0.013       0
G1       51   61    2    2.00      350   0.014   0.000   0.000   0.013       0
G1       61   71    2    3.00      600   0.018   0.000   0.000   0.013       0
G1       44   53    2    1.50      400   0.010   0.000   0.000   0.013       0
G1       53   62    2    1.50      300   0.014   0.000   0.000   0.013       0
G1       62   71    2    1.50      350   0.022   0.000   0.000   0.013       0
G1       71   81    2    3.50      400   0.011   0.000   0.000   0.013       0
G1       81   91    2    3.50      500   0.011   0.000   0.000   0.013       0
G1       91   10    2    3.50      600   0.012   0.000   0.000   0.013       0
G2        0    0    0    0.00        0   0.000   0.000   0.000   0.000       0
H1  1  101   11  600   2.3 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  102   22  600   2.4 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  103   33  600   2.2 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  104   12  330   1.5 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  105   32  650   2.2 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  106   42 2600   9.1 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  107   23 3200  10.2 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  108   34 2000   5.7 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  109   43 4000  11.9 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  110   31 4600  11.1 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  111   41 2200   5.5 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  112   51 1600   4.3 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  113   52 5600  15.4 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  114   61 1600   3.2 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  115   44 2800   8.0 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  116   53 1600   2.9 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  117   62 2600   7.8 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  118   71 1600   4.0 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  119   81 2400   5.5 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H1  1  120   91 2400   5.0 70.00.0020.0170.0250.0500.2003.00 0.30 0.001500
H2  0    0    0    0   0.0  0.00.0000.0000.0000.0000.0000.00 0.00 0.000000
M1 20    1     0     0     0     0     0     0     0     0     0     0
M2 11   22    33    12    32    42    23    34    43    31    41    51    52    61    44    53
M2 62   71    81    91     0     0     0     0     0     0     0     0     0     0     0     0
ENDPROGRAM
```

Figure 5-6. Example Input File for Runoff Module with Designed Pipes in Gutter Cards compared to the previous input file (Figure 5-4) in which the "G1" cards contained only the pipe linkages with the "dummy" gutter option (option "3") invoked. Notice that all other parameters on the "G1" card are zero with this option. The new simulation acts as a verification of the pipe network design.

The peak flows in the design pipe network for the first design step of the example are also shown in Table 5-5. When compared with the peak flows found with the "dummy" gutter system, several values are slightly lower. It will more often be the case that the peak flows will decrease while routing the hydrographs through a real system, although it is possible for the peaks to increase. If the peaks were to increase, two possibilities exist. First, despite the increase, the designed pipe was still sufficient to handle the peak flows. In this case no change need be made in that pipe design. The second possibility is that surcharge occurs in which case SWMM provides a surcharge message in the output file. In the case of the example problem, the peak flows decreased in several of the pipes in the system. These decreases are large enough to merit returning to the drainage design template to re-evaluate the system. The re-evaluation consists of entering the new peak flows into the template and attempting to decrease pipe sizes and/or slopes to arrive at a better (i.e. less-costly) design. In the example, the decreases in peak flows in the second simulation allow decreases to be made in three pipes in the system as shown in the second design columns in Table 5-5.

This new design must next be verified by entering the new system design back into the Runoff Module simulation. This procedure is performed in the same manner as before and the resulting peak flow in each pipe is shown in the last column of Table 5-5. In all cases the new peak flows are very close to those found in the last simulation. In two of the pipes the peak flows did increase very slightly, but no surcharge message was given in the SWMM output so the present design is adequate. Another attempt could be made at this point to reduce pipe sizes, but it is likely that this effort would be futile. The last Runoff Module simulation has, however, verified the last system design.

Conclusions

This linkage of a widely-used and powerful hydrologic model with a simple and yet effective drainage design method offers an advancement toward presenting a relatively simple method of applying complex models to design. In the past, advanced modeling technology and practiced design techniques have existed in parallel with very little interaction. The spreadsheet has been shown to be an effective tool for initializing a link between modeling and design. The prototype described in this section is primitive in that an external database was necessary to link the spreadsheet templates. Although this linkage is relatively easy to perform, it is a manual procedure which is executed by the user. The iterative procedure is also manual in that the user is required to view the Runoff Module output file and extract the peak flow values after each run. These values must also be manually compared to the previous values and entered into the design template.

Ideally, the preprocessor and the design template would be combined so that no external linkage would be necessary. Another idealization entails the complex model being executed as an external macro program which is controlled by the host spreadsheet. With this facility, the spreadsheet is automatically linked with the complex model (e.g. SWMM) which is written in a more powerful programming language to execute efficiently. Spreadsheet software is currently available with this capability. Large computational capabilities are made available while the desirable characteristics of the spreadsheet such as easy input, output, editting, and graphics capabilities are retained. The ability of the spreadsheet to provide easy user interaction is a great asset for its use as a design tool. The prototype presented in this section is a useful example of the ability to apply complex model to design. Although the linkages and procedural steps are somewhat clumsy, the methodology presented is applicable for use with software tools yet to be analyzed. The methodology for the linkage of the complex model and the design template was performed not only as a replacement for the old methods of estimating runoff volumes or performing flow routing (e.g. Rational Method). It is also presented to make more practical the use of simulation such as the modeling of water quality which has not historically been a part of drainage design methods.

CHAPTER 6
SUMMARY AND CONCLUSIONS

Spreadsheet Tabulation Method

Development of the spreadsheet tabulation method was prompted by the discovery that many highway drainage design calculations are still being performed manually. The spreadsheet procedure provides a computerized method of performing drainage calculations in a manner that will facilitate the transition to a computerized environment. The deterrents of using the computerized algorithms presently available are their tedious input procedures, the difficulty in understanding their algorithms, and the difficulty in conforming the problem constraints to the format needed by the algorithm.

The proposed spreadsheet method addresses these deterrents. Although the spreadsheet is not the most efficient tool on which to program this procedure, it provides a user interface capability that is useful for design purposes. The tabulation form may be presented in the spreadsheet in a format that is recognizable to those familiar with the tabulation procedure. The formula used in each computation of the procedure may be seen and checked by the user without having to search computer code. The spreadsheet procedure also allows the engineer to employ personal strategies and judgement. All of these features give the engineer confidence in the final design acquired with the use of the spreadsheet tabulation form.

Spreadsheet Procedure vs. Dynamic Programming

Not only does the spreadsheet provide a familiar format for performing drainage designs, but this procedure is capable of producing designs that are better than those derived using advanced dynamic programming algorithms. The problem constraints are easily defined, and no simplifying assumptions are necessary in order to apply the algorithm to the problem as is common with optimization algorithms. The spreadsheet template does not employ advanced mathematical algorithms in its design procedure. The template is only a method of electronically tabulating the design procedure that is currently being utilized by the Florida Department of Transportation.

Because the spreadsheet template does not employ rigorous mathematical algorithms which consume computer time, it can employ more refined design calculations. For example, the simplified velocity calculations employed by the analyzed dynamic programming algorithms resulted in designs which did not meet all of the defined problem constraints. These dynamic programming algorithms applied the majority of their computer power to the solution of the mathematical optimization algorithm rather than to more adequate design calculations. The spreadsheet algorithm also allows the engineer to consider criteria that may not be defined in terms of cost. Designs that provide alternative benefits may be considered without modification of the algorithm as would be necessary with partial enumeration algorithms.

Itemized Cost Estimates as a Design Heuristic

A cost estimation scheme is essential to a design procedure in order to ensure that the design is economically as well as technically feasible. In drainage design, cost is especially important because of the trade-offs between pipe costs and excavation costs that are found. The application of rules of thumb in a design are helpful in reducing the system cost, but are poor substitutes for a computed cost estimate of a system.

Many computerized design algorithms have used cost estimating functions to incorporate cost considerations into their designs. The cost functions may be a function of one or two parameters such as pipe diameter and/or invert depth. Despite the number of many parameters they may be based upon, these functions cannot relate the trade-offs found in a problem as accurately as an itemized cost estimate. The spreadsheet template provides an itemized cost estimate during the design process. The cost estimate is based on the FDOT itemized average bid cost database for highway construction items. Access to this information during the design process allows the engineer to employ a heuristic approach in finding the least-cost system design.

The FDOT itemized cost database is accessed with the use of an external add-on feature available with Lotus 1-2-3. The database is contained in an external file which can be accessed from any spreadsheet file. This capability saves greatly on storage space in the spreadsheet file. This feature also facilitates the updating of the database.

Linking of the Drainage Design Template and SWMM

The external database capability is also used as a link between the drainage design template and a spreadsheet based preprocessor for a personal computer version of the SWMM Runoff Module. The Runoff Module is used as an advanced method of determining the peak flow capacities required in the system for a given design rainfall event. The Runoff Module is also used as a verification of the design procedure.

Because of the batch run format of the Runoff Module, the design procedure is implemented iteratively. The Runoff Module is run with "dummy" gutters in place of the pipe system to determine preliminary design flows in the system. With these preliminary flows, a system is designed using the spreadsheet design template. The "dummy" gutters are next replaced by the new system design and the Runoff Module is run as a verification of the design. If the design flow capacities of the pipes change in this simulation, then the pipe design is re-evaluated in the design template. This procedure is repeated until a system design is verified to be feasible and is judged to be the least-cost design.

The spreadsheet preprocessor provides a simple method of creating the required formatted input file for the SWMM Runoff Module. Input parameters are entered directly next to the parameter name and its brief description. Users are also offered more extensive help files when necessary for determining a parameter value.

Conclusions

The spreadsheet drainage design procedure is a proposed alternative to the manual drainage design method presently used by the FDOT. The spreadsheet template has been constructed to mimic the calculation procedure performed on the FDOT tabulation form. The ability to employ itemized cost estimates of the system design as a heuristic is also featured on the spreadsheet template. Although this procedure is based on straight-forward design procedures, it has been shown to produce better designs than advanced optimization algorithms. It has also been shown that the spreadsheet procedure is capable of producing designs that are less-costly than those produced with the current FDOT design procedures on a real-world problem.

The only challenge yet to be met by the spreadsheet procedure is its use by current drainage engineers. The current organization of the spreadsheet design template is certainly imperfect and modifications will need to be made to facilitate its use by others. However, the important concepts have been addressed and its potential to be a powerful and efficient design algorithm has been established. It is hoped that the ideas presented in this thesis will be applied to improve the drainage design procedures currently in use.

Suggestions for Additional Research

Further research along the lines of this thesis may explore more deeply the accuracy of the detailed cost estimating procedure. The current drainage design template does not allow for the incorporation of cost factors to account for the uniqueness of projects. For example, the excavation costs may contain a factor for the amount of blasting needed or the previous land use. The location of the project may account for adjustments in material costs because of shipping or hauling costs. Many such factors are often included by cost estimators in their detailed estimates. An expert system may also be included to assist engineers in choosing these factors based upon predetermined rules.

Another component of the cost estimation which may be included is a risk analysis of the reliability of the design. As mentioned, this capability is already available in design algorithms to determine the risk involved with the failure of an impending design. This ability would provide much insight to the drainage engineer as to the trade-offs contained in the drainage system design.

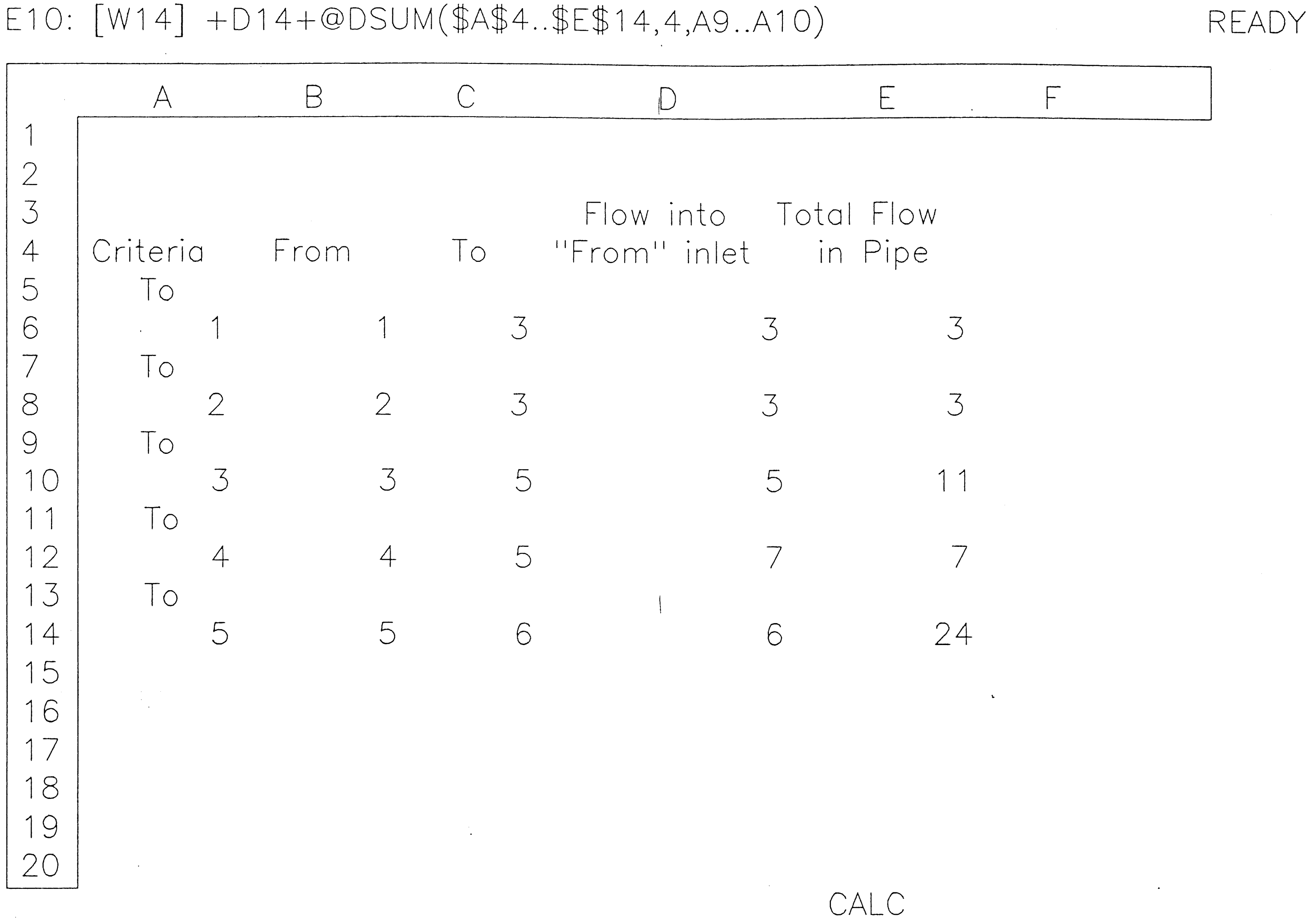
Lastly, the crude linkage between the design template and a complex hydrologic model should be improved upon. The ability of the spreadsheet to easily access an external programming language would make the spreadsheet an even more powerful tool for design purposes.

APPENDIX

SPREADSHEET REPRESENTATION OF PIPE SYSTEM LAYOUT

An important concept used throughout this design template is the manner in which the pipes are electronically linked. This concept must be used when calculating the elevations of the pipe inverts and the hydraulic grade line through the system. For example, if the hydraulic grade line at the upstream end of a pipe is at an elevation of 100 ft., then the hydraulic grade line at the downstream end of all pipes flowing into the first pipe must also be 100 ft. The ability to connect pipes on the basis of their defined (to and from) nodes must be performed with the pipes being entered in no particular order in the spreadsheet template.

This ability is performed in the spreadsheet template with the use of the Lotus 1-2-3 database functions. These functions allow the extraction of specific information which meets defined criteria from a data range. In order to describe the application of these functions to the linkage problem, it is easiest to present a simple example problem. A simple pipe network is defined (see Figure A-1) by the to and from nodes of the pipes. In this description, all pipes with a "to" node of "3" flow into the pipe with a "from" node of "3." Any number of pipes may flow into a node; however, the definition of the problem only allows one pipe to flow from any node. In the example, inlet flows at the upstream end of each pipe are given. The total flow column must



```
E10: [W14] +D14+@DSUM($A$4..$E$14,4,A9..A10)                     READY
        A           B           C              D              E          F
 1
 2
 3                                         Flow into     Total Flow
 4   Criteria     From         To        "From" inlet     in Pipe
 5     To
 6         1           1        3                3            3
 7     To
 8         2           2        3                3            3
 9     To
10         3           3        5                5           11
11     To
12         4           4        5                7            7
13     To
14         5           5        6                6           24
15
16
17
18
19
20
                                                     CALC
```

Figure A-1. Illustration of Pipe Linkage in Spreadsheet calculate the flow in each pipe based on the inlet flows of all upstream pipes.

In the example problem there are two pipes flowing into node 3. The flows in these two pipe are 2 cfs and 4 cfs. Therefore, the total flow column must add these two flows to the inlet flow of 5 cfs at node three to give a total flow of 11 cfs through pipe 3-5. The top of Figure A-1 shows the formula contained in the cell which calculates the flow through pipe 3-5. The general format of the @dsum function is as follows:

@dsum(data range, column offset, criteria range)

The data range is the entire range in which data may be found including the row of titles at the top of the data. The titles are needed in this range for Lotus to match the criteria. The criteria range consists of a column title and criteria. The criteria may include "greater than," "less than," etc., but in this case only an exact match is needed so only the number to be matched must be entered under the criteria title. The column offset is the column number of the number to be returned by the function. In this example the inlet flows are in the third column over from the first, and so the column offset equals three. Therefore, the "@dsum" formula at the top of Figure A-1 returns the sum of all numbers in third column that meet the criteria that the "to" node is "3." This result, when added to the inlet flow at node "3," produces the desired answer of 11 cfs. This method of matching "to" and "from" nodes in used throughout the drainage design template for matching invert elevations and hydraulic grade lines along with other applications.

REFERENCES

American Society of Civil Engineers and Water Pollution Control Federation (1969). Design and Construction of Sanitary and Storm Sewers, ASCE Manual 37, New York, NY.

Arnell, V. (1982). "Rainfall Data for the Design of Sewer Pipe Systems." Report Series A:8, Dept. of Hydraulics, Chalmers University of Technology, Goteborg, Sweden.

Austin, T.A. (1986). "Utilization of Models in Water Resources." Water Resources Bulletin, Vol. 22, No. 1, pp. 49-53.

Brown, K.G. and A.D. Koussis (1987). "LOTUS Spreadsheet Design for Storm Drain Networks." J. of Computing in Civil Engineering, ASCE, Vol. 1, No. 3, pp. 197-213.

Christensen, B.A. (1984). "Analysis of Partially Filled Circular Storm Sewers." In: Proceedings of Water for Resource Development. Hydraulics Division, ASCE, New York, NY, pp. 163-167.

Cunningham, B.A. (1987). Economic and Predictive Reliability Implications of Stormwater Design Methodologies. Master's Thesis, University of Florida, Gainesville, FL.

Dennison, K.D. and W. James (1985). "A Database Environment and Sensitivity Framework for a Continuous Water Quality Model." In: Proceedings of Conference on Stormwater and Water Quality Management Modeling, McMaster University, Hamilton, Ontario. U.S. Environmental Protection Agency, Athens, GA, pp. 101-123.

Desher, D.P. and P.K. Davis (1986). "Designing Sanitary Sewers with Microcomputers." J. of Environmental Engineering, ASCE, Vol. 112, No. 6, pp. 993-1007.

Dodge Cost Systems (1987). Dodge Unit Cost Data, McGraw-Hill, Princeton, NJ.

Drainage Manual (1987), Florida Department of Transportation, Tallahassee, FL.

Fedra, K. and D.P. Loucks (1985). "Interactive Computer Technology for Planning and Policy Making." Water Resources Research, Vol. 21, No. 2, pp. 114-122.

Grigg, N.S. and J.P. O'Hearn (1976). "Development of Storm Drainage Cost Functions." J. of the Hydraulics Division, ASCE, Vol. 102, No. HY4, pp. 515-523.

Han, J., A.R. Rao, and M.H. Houck (1980). "Least Cost Design of Urban Drainage Systems." Purdue University Water Resources Research Center Technical Report No. 138, West Lafayette, IN.

Hancock, M.C. and J.P. Heaney (1987). "Water Resources Analysis Using Electronic Spreadsheets." J. of Water Resources Planning and Management, ASCE, Vol 113, No. 5, pp. 639-658.

Hillier, F.S. and G. J. Lieberman (1986). Introduction to Operations Research, Fourth Edition, Holden Day, Inc., Oakland, CA.

Holland, M.E. (1966). Computer Models of Wastewater Collection Systems. Thesis, The Division of Engineering and Applied Physics, Harvard University, Cambridge, MA.

Huber, W.C., J.P. Heaney, S.J. Nix, R.E. Dickinson, and D.J. Polmann (1981). "Stormwater Management Model User's Manual, Version III." EPA-600/2-84-109a (NTIS PB84-198423). U.S. Environmental Protection Agency, Cincinnati, OH.

James, W., and M. Robinson (1985). "PCSWMM3.2 User's Manual," Computational Hydraulics, Hamilton, Ontario.

James, W., M. Robinson, and M. Stirrup (1986). "PC Software for Computational Hydrology." In: Proceedings of Stormwater and Water Quality Model Users Group Meeting, Orlando, FL. EPA/600/9-86/023. U.S. Environmental Protection Agency, Athens, GA, pp. 45-57.

Klemes, V. (1979). "Storage Mass-Curve Analysis in a Systems-Analytic Perspective" Water Resources Research, Vol. 15 No. 2, pp. 359-370.

Labadie, J. W. (1987). "Dynamic Programming with the Microcomputer." Encyclopedia of Microcomputers, Marcel Dekker, Inc., New York.

Liebman, J.C. (1967). "A Heuristic Aid for the Design of Sewer Networks." J. of the Sanitary Engineering Division, ASCE, Vol. 93, No. SA4, pp. 81-90.

Linsley, R.K. (1986). "Flood Estimates: How Good Are They?" Water Resources Research, Vol. 22, No. 9, pp. 159S-164S.

Long Range Estimating Manual (1986), Florida Department of Transportation, Tallahassee, FL.

Lotus Development Corp. (1986). Lotus 1-2-3 User's Manual, Release 2.01. Cambridge, MA.

Mays, L.W. and H.G. Wenzel (1976). "Optimal Design of Multilevel Branching Sewer Systems." Water Resources Research, Vol. 12 No. 5, pp. 913-917.

Meredith, D.D. (1972). "Dynamic Programming with Case Study on Planning and Design of Urban Water Facilities." Treatise on Urban Water Systems, Colorado State University, Fort Collins, CO, pp. 37-47.

Merritt, L.B., and R.H. Bogan (1973). "Computer-Based Optimal Design of Sewer Systems." J. of Environmental Engineering Division, ASCE, Vol. 99 No. EE1, pp. 35-53.

Miles, S.W. (1986). "Spreadsheet Drainage Design." Unpublished presentation to Florida Department of Transportation, Tallahassee, FL.

Miles, S.W., J.P. Heaney, W.C. Huber, and T.G. Potter (1986). "Application of Lotus 1-2-3 to SWMM Preprocessing," In: Proceedings of the Fourth Conference on Computing in Civil Engineering, Boston, MA. ASCE, New York, NY, pp. 969-984.

Miles, S.W., T.G. Potter, and J.P. Heaney (1987). "The Use of Detailed Cost Estimation for Drainage Design Parameter Analysis on Spreadsheets." In: Proceedings of Stormwater and Water Quality Model Users Group Meeting, Denver, CO. EPA/600/9-87/016, U.S. Environmental Protection Agency, Athens, GA, pp. 180-193.

Personics Corporation, (1987). "@Base User's Manual." Concord, MA.

Potter, T.G. (1986) "PC Version of FDOT Drainage Model." Unpublished presentation to Florida Department of Transportation, Tallahassee, FL.

Robinson, D.K. and J.W. Labadie (1981). "Optimal Design of Urban Storm Water Drainage Systems." International Symposium on Urban Hydrology, Hydraulics, and Sediment Control. University of Kentucky, Lexington, KY, pp. 145-156.

Roesner, L.A., R.P. Shubinski, J.A. Aldrich (1983). "Stormwater Management Model User's Manual Version III, Addendum I Extran." EPA-600/2-84-109b (NTIS PB84-198431). U.S. Environmental Protection Agency, Cincinnati, OH.

Simpson, A. (1986). Simpson's 1-2-3 Macro Library, Sybex Inc., Berkeley, CA.

Smith, A.A., E. Hinton, R.W. Lewis (1983). Civil Engineering Systems Analysis and Design. John Wiley and Sons, New York, NY.

South Florida Water Management District, (1987). "Management and Storage of Surface Waters--Permit Information Manual, Volume IV." West Palm Beach, FL.

Tang, W.H., L.W. Mays, and B.C. Yen (1975). "Optimal Risk-Based Design of Storm Sewer Networks." J. of the Environmental Engineering Division, ASCE, Vol. 101, No. EE3, pp. 381-397.

Thompson, G.R., and B.C. Phillips (1987). "Microcomputers -- The Stormwater Modelling Future," In: Proceedings of Stormwater and Water Quality Model Users Group Meeting, Denver, CO. EPA/600/9-87/016. U.S. Environmental Protection Agency, Athens, GA, pp. 10-20.

Tyteca, D. (1976). "Cost Functions for Wastewater Conveyance Systems." J. of Water Pollution Control Federation, Vol. 48, No.9, pp. 2120-2130.

Walski, T.M. (1980). "MAPS--A Planning Tool for Corps of Engineers Regional Supply Studies." Water Resources Bulletin, Vol. 16, No. 2, pp. 215-219.

Whipple, W., N.S. Grigg, T. Grizzard, C.W. Randall, R.P. Shubinski, and L.S. Tucker (1983). Stormwater Management in Urbanizing Areas, Prentice-Hall, Inc., Englewood Cliffs, NJ.

Yen, B.C., S. Cheng, B. Jun, M.L. Voorhees, H.G. Wenzel, Jr., and L. W. Mays (1984). Illinois Least-Cost Sewer System Design Model: ILSD-1&2 User's Guide. University of Illinois Water Resources Center Research Report No. 188.

Yen, B.C., H.G. Wenzel, Jr., L.W. Mays, and W.H. Tang (1976) Advanced Methodologies for Design of Storm Sewer Systems. University of Illinois Water Resources Center Research Report No. 112.

Zepp, P.L. and Leary, A. (1969). A Computer Program for Sewer Design and Cost Estimation. Regional Planning Council, Baltimore, Maryland, available as PB 185 592, Clearinghouse, U.S. Dept. of Commerce, Springfield, VA.

BIOGRAPHICAL SKETCH

Stewart Wayne Miles was born on July 24, 1963, in Poughkeepsie, New York to Donald and Sally Miles. He has two younger brothers, Kevin and Scott. He attended public schools and graduated from Arlington High School, Pleasant Valley, New York, in the Spring of 1981 with a concentration on the study of the feeding habits of the largemouth bass. He later turned his study southward by attending the University of Florida where he received his Bachelor of Science degree in environmental engineering in the Spring of 1986. He continued to attend the University of Florida after graduation with a focus on water resources and received a Master of Engineering degree in April, 1988. His plans include moving to Raleigh, North Carolina, to begin his professional career. He is engaged to be married to the girl of his dreams, Jessica Godreau, on September 3 of this year.