A Computerized Master Plan of Drainage, II: Software System

T.V. Hromadka II

Director of Water Resources Engineering, Williamson and Schmid, 17782 Sky Park Blvd., Irvine, CA 92714, USA

Abstract

In this paper, the computerized master plan of drainage software system is presented. Composed of several subroutines linked together by a main-menu driver, the program is intended to be operated in an interactive mode where the program user accepts or rejects all analysis results as the data file is developed. In this fashion, an optimum design product is developed in the first data entry pass.

Keywords

Water Resources; Hydrologic Modeling; Master Plan of Drainage; Urban Drainage

A Rational Method Planning/Design Computer Program

Each of the rational method modeling approaches (Hromadka, 86) utilize identical submodels for estimating (1), the initial time of concentration, (2), channel or pipeflow traveltime, (3), runoff coefficients, (4), rainfall intensity values, (5), and confluence values at the junction of two or more collection streams. Therefore, since computer program subroutines are developed for each of these submodels, a main driver program can be developed which manipulates the individual submodels to formulate a link-node model of the watershed based on the rational method modeling strategy desired. In Table 1, descriptions are listed for the computer programs used to model the hydrologic processes which occur in a rational method study of an urban watershed. Combining these programs using a simple main menu which branches the program to the selected submodel will result in a totally design-interactive computer program.

<table>
<thead>
<tr>
<th>Program Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>main driver program</td>
</tr>
<tr>
<td>2</td>
<td>utilizes the Kirschbach formula for estimating the initial subarea time of concentration Tc</td>
</tr>
<tr>
<td>3 (no input)</td>
<td>calculates rainfall intensities by log-log interpolation</td>
</tr>
<tr>
<td>4 (no input)</td>
<td>estimates a runoff coefficient</td>
</tr>
<tr>
<td>5</td>
<td>estimates pipeflow traveltime for a user-specified or computer estimated pipe size</td>
</tr>
<tr>
<td>6</td>
<td>estimates traveltime in a trapezoidal channel</td>
</tr>
<tr>
<td>7</td>
<td>estimates traveltime in a street section of arbitrary size</td>
</tr>
<tr>
<td>8</td>
<td>estimates traveltime in a pavement gutter</td>
</tr>
<tr>
<td>9</td>
<td>estimates confluence values</td>
</tr>
<tr>
<td>10</td>
<td>allows entry of user-specified data at a node</td>
</tr>
</tbody>
</table>

Programs 2, 3, and 4 follow directly from material presented above in the text and the referenced figure numbers. Program 5 estimates pipeflow traveltime by computing the normal depth and determining the time of travel based upon the normal depth flow velocity. Flows which result in a normal depth greater than 0.8 of the pipe diameter are assumed to cause the pipe to flow full. If the pipe size is not specified, this program estimates a pipe size in 3- and 6-inch increments by utilizing a pipeflow with a normal depth less than or equal to 0.8 of the pipe diameter. Pipe slope is based on ground slope; however, a factor is introduced such that the natural gradient of the land is reduced (usually by about 10 percent) in order to account for minor losses within the pipe. The pipe sizes are estimated by assuming the adjusted gradient of the topography between two nodal points to equal the slope of the pipe for normal depth flows. Program 6 estimates channel flow traveltime based upon the normal depth flow velocity. Program 7 examines streetflow traveltime for two conditions; (1), all flow on one side of the street section, including the splitter effects when the flowdepth exceeds the street crown, and (2), equal flow on both sides of the street section. All flows outside of the street curbs are assumed negligible (that is, that water is in a ponded condition). Program 8 selects a confluence with up to 2 independent collection streams. It is based upon the linear confluence formula presented in the text.

The usual study approach is to subdivide the watershed into subareas such as shown in Fig. 1. Nodal points are defined at the upstream and downstream points of each subarea. Computer results are correlated to the hydrology map by means of these nodal point designations. The programs are intended to be combined into a menu-driven program system in which the user interacts with the program. Starting at the most upstream nodal point of a collection stream, the program user selects which submodel is to be first employed. Usually, the first model is the initial subarea program and the user enters the appropriate hydrologic data such as subarea development type, soil group, area size, upstream and downstream elevations, and length of the main flowpath. The submodel computes the initial subarea Tc, the corresponding runoff coefficient, and rainfall intensity, and the initial subarea runoff. The program user then is displayed this information for the user to accept or reject. If the information is acceptable, the entered hydrologic data is permanently stored in a data file; if the computed results are unacceptable, the user rejects the submodel results and the computer program returns to the previous nodal point.

If the user had accepted the most recently computed information, then the main program returns to the menu display for the user to select the next hydrologic submodel. The main program should store the recently computed Q, Tc and the total area. In
this manner, should the user now select to employ the channel selection program, the default option will be based on the stored peak Q value, and the traveltime will be directly added to the stored Tc value, providing the time of concentration at the downstream point in question. Thus, the computer program directly follows the desired rational method modeling approach interactively rather than the user creating a data file to be operated upon by the procedure. Such a computer programming approach allows the watershed to be master planned on the first study pass, and in addition, the entered hydrologic data are stored for subsequent editing and master plan updates.

In the following pages, computer listings are provided for each of the discussed submodels. The language used is FORTRAN, and the code is directly usable on many currently available microcomputers. The data entry requirements are presented in the form of screen text pages which contain the submodel data entry prompts as well as other user-friendly program commands and features. Details of these screen text pages are discussed in the following section.

Computer-Aided Design Interaction

The computer programs were developed to aid the engineer in a computer-aided interactive mode rather than the inefficient and difficult to use batch mode that is associated to most water resources software. In this fashion, the software is formulated on the level where the individual submodels are employed as selected by the engineer, and the computed results reviewed by the engineer prior to proceeding with the next submodel process. This type of approach can be directly applied to link-node models where the links direct the logic process in one direction only. For example, the rational method planning/design program process is performed in the downstream direction with the entire watershed tributary to a node completely described by three variables: peak runoff rate, time of concentration, and total area. Thus the hydraulic process employed to link the next downstream node acts only upon the most recently computed values of the three characteristic variables. Because the main purpose of studying the watershed is to determine an appropriate flood control system to safely contain the peak flow rates, each link of the link-node model is properly sized and evaluated prior to proceeding to the next link or hydraulic process.

In comparison, the various submodels can be combined into a batch mode of operation where the engineer builds a data file containing all the necessary data for each hydrologic process or link used to develop a link-node model of the watershed. The program system then operates upon the data file to generate the model solutions. The user then reviews the entire model results for unacceptable conditions (e.g., such as streamflow above the top of curb, or excessively high flow velocities in a user-specified pipe size linking two model points, etc.) and identifies the necessary alterations in the link-node model to remedy the computer design. This procedure is repeated until the entire link-node model provides the required flood control system design, and results in a considerable amount of computational effort, time expenditure, and frustration to the engineer.

Therefore, the engineer should develop the main driver branching program using the basic user-friendly environment as discussed in Hromadka et al. (1983c). The main program data entry sequences for each submodel should be developed such that the communication/presentation (C/P) provides an easy-to-use and self-teaching environment. Some of the major requirements for such a user-friendly environment are as follows:

1. The C/P should present all data entry prompts and computed results in a readable manner such that any engineer can readily evaluate the information.
2. All engineering units should be displayed.
3. Any program system flow logic should be clearly described in the program where needed in order to reduce the first-time user learning curve.
4. All program system commands should be consistently displayed between submodels (and between separate computer programs) so that new data files can operate separate data entry or editing features without confusion.
5. All data file management operations (such as opening, closing, and saving data files) should be programmed interior to the system program in order to provide ease of use.

The submodel data entry prompts for the provided program listings are presented in a typical C/P for use on currently available microcomputers. The viewing displays are constructed as pages which contain sets of data entry prompts grouped together according to the selected submodel process. Each of the pages contain the following set of operation commands located at the bottom of the CRT screen:

1. TOP: This command clears the screen, rediscards the previous page information, and returns the program to the first data entry prompt of the page.
2. BACK: This command returns the program to the previous page (if one exists), and positions the program to the first data entry prompt.
3. MAIN: This command performs several important tasks. First, the program system data file is properly saved and closed so that all data entries are protected, and the data file is available for later use. Second, the command terminates the submodel process in progress should the user be interior of a subrouting process. Third, the command returns the program system to the main driver program menu.
4. EXIT: This command is identical to the MAIN command, except the program system is terminated.

It should be noted that these four commands can be entered at any time, and at any data entry prompt within the program system. Thus, if the user should wish to exit the program while entering the data needed to solve for pipeflow between two model points (Program 5), then the user simply enters the word EXIT at any data entry prompt. It should also be noted that the C/P pages contain a description of each data entry as well as the allowable value range for data entry. Each data entry is checked for range limits prior to proceeding to the next data entry prompt. If the entered data is outside of the allowable value range, an error message is displayed to the user and the program returns to the invalid data entry point for another data entry attempt. In this way, the data file development is error free with the first pass of the data entry sequence.

Unfortunately, there are still wide differences between computers and peripheral devices such as the CRT terminals of different manufacturers. Additionally, the internal operating systems of the computers differ. Developing a user-friendly environment for the computer program system is therefore dependent on the computer system hardware and associated software. The necessary steps in opening files, closing files, subroutine
employment, overlay structuring, cursor addressing, clearing the screen, program execution and other operations must be obtained from the computer user-guides and operation manuals for each particular system.

Example 1. Rational Method Program Application

The following computer program application example problem illustrates the use of the Subarea Summary Model for rational method hydrology studies of urban watersheds. The example problem presentation contains the following information:

Figure Number Description
1 Example problem drainage system
2 Example problem point rainfall
3 Example problem computer program results and example tabulation form output

References


SOIL CLASSIFICATION IS "A"
SINGLE-FAMILY (1.25 ACRE LOT) RUNOFF COEFFICIENT = 0.35
SUBAREA AREA (ACRES) = 6.86 SUBAREA RUNOFF (CFS) = 7.77
END OF SUBAREA DRAINAGE HYDRAULICS
DEPT OF FLOW (FEET) = 0.25 FLOW VELOCITY (FEET/SEC.) = 1.89
STREAM RAINFALL INTENSITY (INCH/HOUR) = 1.44

FLOW PROCESS FROM NODE 21.00 TO NODE 14.00 IS CODE = 1

FLOW PROCESS FROM NODE 15.00 TO NODE 14.00 IS CODE = 5

FLOW PROCESS FROM NODE 15.00 TO NODE 14.00 IS CODE = 5

FLOW PROCESS FROM NODE 21.00 TO NODE 14.00 IS CODE = 1
END OF NATIONAL METHOD ANALYSIS
IF (ITER.EQ.1) WRITE(NUT,3120)

1120 FORMAT(ILO,1,'TRAVELTIME COMPUTED USING MEAN FLOW(CTS) = ',PG,T2.2)
       ERROR=0
       T2=QTR/CP-1.*T6+H2*Q/
       TN=1.066/4*T3*Q**.6667
       HS-QTR-1*Q6
       HS-QTR-1*Q6+T6-4*4*Q6
       HS-QTR-1*Q6+T6-4*4*Q6
       TN=1.066/4*T3*Q**.6667
       D=Q/
       IF (WIDTH.EQ.2) D=W2/2.
       D=W2/2.
       IF (WIDTH.EQ.3) D=W2/2.
       V=Q/A1
       V65/2+Q6/2
       V=Q/A1
       V=W2+Q6/2
       IF (WIDTH.EQ.3) WRITE(NUT,3120)
       IF (WIDTH.EQ.2) WRITE(NUT,3120)
       IF (WIDTH.EQ.1) WRITE(NUT,3120)

300 FORMAT(ILO,1,'FLOW IS LESS THAN CROWN')
       TEST=0.
       TN=H2+Q6/2
       TEST=H2+Q6
       IF (WIDTH.EQ.3) WRITE(NUT,3120)
       IF (WIDTH.EQ.2) WRITE(NUT,3120)
       IF (WIDTH.EQ.1) WRITE(NUT,3120)

350 FORMAT(ILO,1,'FLOW EXCEEDS CROWN')
       IF (WIDTH.EQ.3) WRITE(NUT,3120)
       IF (WIDTH.EQ.2) WRITE(NUT,3120)
       IF (WIDTH.EQ.1) WRITE(NUT,3120)

400 CONTINUE

450 END

C--------FLOW SPLITS AND IS LESS THAN FULL(CROWN) STREET
       Q6=Q/3
       Q6=Q/3
       IF (WIDTH.EQ.3) WRITE(NUT,3120)
       IF (WIDTH.EQ.2) WRITE(NUT,3120)
       IF (WIDTH.EQ.1) WRITE(NUT,3120)

435 FORMAT(ILO,1,'STREETFLOW SPLITS OVER STREET-CROWN')
       Q6=Q/3
       Q6=Q/3
       IF (WIDTH.EQ.3) WRITE(NUT,3120)
       IF (WIDTH.EQ.2) WRITE(NUT,3120)
       IF (WIDTH.EQ.1) WRITE(NUT,3120)

477 FORMAT(ILO,1,1,'FLOW IS LESS THAN CROWN')
       GS=Q6
       GS=Q6
       IF (WIDTH.EQ.3) WRITE(NUT,3120)
       IF (WIDTH.EQ.2) WRITE(NUT,3120)
       IF (WIDTH.EQ.1) WRITE(NUT,3120)

430 FORMAT(ILO,1,1,'SUMMARY')
       Q6=Q/3
       GS=Q6
       IF (WIDTH.EQ.3) WRITE(NUT,3120)
       IF (WIDTH.EQ.2) WRITE(NUT,3120)
       IF (WIDTH.EQ.1) WRITE(NUT,3120)

441 FORMAT(ILO,1,1,1,'FLUSHING THE STREE')
       GS=Q6
       GS=Q6
       IF (WIDTH.EQ.3) WRITE(NUT,3120)
       IF (WIDTH.EQ.2) WRITE(NUT,3120)
       IF (WIDTH.EQ.1) WRITE(NUT,3120)

445 FORMAT(ILO,1,1,1,1,'SUMMARY')
       GS=Q6
       GS=Q6
       IF (WIDTH.EQ.3) WRITE(NUT,3120)
       IF (WIDTH.EQ.2) WRITE(NUT,3120)
       IF (WIDTH.EQ.1) WRITE(NUT,3120)
PROGRAM 10: DATA ENTRY

---DATA ENTRY FOR SPECIFICATION OF HYDROLOGY DATA AT A NODE---

Enter user-specified time of concentration (i.e., ...) = "".
(INVALID VALUES ARE [1] TO [1000] i)

Enter user-specified total area (i.e., ...; tributary to node) = "".
(INVALID VALUES ARE [1] TO [1000] i)

Enter user-specified total precipitation = "".
(INVALID VALUES ARE [1] TO [1000] i)

Fig. P.9.1

MAIN-LINE CONFIGURATION REQUIRING ONE CONFLUENCE

DO 590 [*]=1, NUMBER
590 CONTINUE

IF (QMAX <= QBAR) QBAR = QMAX
QBAR = (QBAR + QBAR) / 2
CONTINUE

WRITE(*,595) J, TC, SUMA

WRITE(*,599) J, TC, SUMA
599 FORMAT(2X, 'TOTAL AREA (ACRES) = ', SUMA, 2X, 'TOTAL RUNOFF (CFPS) = ', TC)

RETURN

END

PROGRAM 10

SUBROUTINE "HYD" (QBAR, TC, SUMA, IERR)

COMMON /QBAR/ QBAR, TC, SUMA

COMPUTE HYDROLOGY INFORMATION AT NODE J.

READ 1, TC, SUMA

WRITE(OUT, 5143) FORMAT(2X, 'USER-SPECIFIED HYDROLOGY INFORMATION AT NODE J.

WRITE(OUT, 509) FORMAT(2X, 'FINAL AREA (ACRES) = ', SUMA, 2X, 'FINAL RUNOFF (CFPS) = ', TC)

RETURN

END

PROGRAM 11: DATA ENTRY

---DATA ENTRY FOR SPECIFICATION OF SUBAREA TO MAINLINE RUNOFF AT MAINLINE TIME OF CONCENTRATION---

SUBAREA LAND USE OR DEVELOPMENT TYPE:

1. Commercial
2. Apartment
3. Mobile home park
4. Industrial
5. Single family/acre
6. Single family/acre
7. Single family/acre
8. Single family/acre
9. Undeveloped

Specify assumed uniform subarea land use/development type = "".

RETURN

END

TYPE: EXIT to leave program; TOP to go to top of page
DATA ENTRY FOR ESTIMATION OF SUBAREA TO MAINLINE RUNOFF

AT MAINLINE TIME OF CONCENTRATION—PAGE 2
SUBAREA RUNOFF COEFFICIENT OPTIONS:

1) Assume soil group A
2) Assume soil group B
3) Assume soil group C
4) Assume soil group D
5) User to specify runoff coefficient

SELECT RUNOFF COEFFICIENT OPTION NUMBER: \( \Rightarrow \"S\" \)

Enter SUBAREA runoff coefficient: \( \Rightarrow \text{VALUE} \)

ALLOWABLE VALUES ARE [.01 TO 1.00]

Enter SUBAREA area (ACRES): \( \Rightarrow \text{AREA} \)

ALLOWABLE VALUES ARE [0 TO 20]

TYPE: Exit to leave program to go to top of page
; BACK to go back one page

SUBROUTINE SUBROUTINE (L, T, S, U, M, F, E, X, Y, Z)

ADDITION OF SUBAREA TO MAINLINE RUNOFF

COMMON /HUT/HUT
COMMON/CH/TV, XV, OPTN, HR, C, CVAL, V, T, T, U, M, F, E, X, Y, Z

COMMON /CUT/TV, XV, HZ, ZS, AREA, XA, AP, HS, ITS, RISE, RISE2

READ DATA INPUT
READ (FZOIL.5) R, FZOIL, AREA
IS=1+FZOIL

C PROCESS SUBAREA ADDITION TO MAINLINE FLOW

WRITE (HUT, 6180)

6180 FORMAT (3X, I6) \"ADDITION OF SUBAREA TO MAINLINE FLOW\"\(\"\"\)
WRITE (HUT, 6182) ERROR, HR, 1
GO TO 6182
ERROR=9
GO TO 3020

6182 CALL COMPAR (X, Y, Z, C, S, CVAL, HUT)

WRITE (HUT, 6200) AREA, DSOIL, SUMA, Q, TC

6200 FORMAT (3X, 'SUBAREA AREA (ACRES) = ', F7.3, 'SUBAREA RUNOFF: ', 
C 'CPS = ', 
C F7.3, 'TOTAL AREA (ACRES) = ', F7.3, 'TOTAL RUNOFF (CPS) = ', 
C F7.3, 'TC (HR) = ', F7.2)

3020 CONTINUE

C FORMATS

403 FORMAT (3X, 14, \"=\")

RETURN

END