8-1990

Hydrology and Sedimentology of Dynamic Rill Networks Volume II: Hydrologic Model for Dynamic Rill Networks

Digital Object Identifier: https://doi.org/10.13023/kwrri.rr.179

Lindell E. Ormsbee
University of Kentucky, lindell.ormsbee@uky.edu

George E. Blandford
University of Kentucky

John S. Montgomery
University of Kentucky

Laura B. Terrell
University of Kentucky

Billy J. Barfield
University of Kentucky

See next page for additional authors

Repository Citation

Ormsbee, Lindell E.; Blandford, George E.; Montgomery, John S.; Terrell, Laura B.; Barfield, Billy J.; and Storm, Daniel E., "Hydrology and Sedimentology of Dynamic Rill Networks Volume II: Hydrologic Model for Dynamic Rill Networks" (1990). *KWRRI Research Reports*. 28.
https://uknowledge.uky.edu/kwrri_reports/28

Authors
Lindell E. Ormsbee, George E. Blandford, John S. Montgomery, Laura B. Terrell, Billy J. Barfield, and Daniel E. Storm

Research Report No. 179

FINAL REPORT

HYDROLOGY AND SEDIMENTOLOGY OF DYNAMIC RILL NETWORKS

VOLUME II:

HYDROLOGIC MODEL FOR DYNAMIC RILL NETWORKS

Lindell E. Ormsbee, George E. Blandford, John S. Montgomery
Laura B. Terrell,

Department of Civil Engineering
University of Kentucky
Lexington, KY 40506

and

Billy J. Barfield, and Daniel E. Storm

Department of Agricultural Engineering
University of Kentucky
Lexington, KY 40546

Final Report from University of Kentucky to U.S. Geological Survey Matching Grant No. 14-08-001-G1147 entitled "Hydrology and Sedimentology of a Dynamic Rill Network"

Project Period 10/1/1985 - 2/28/1990

August 1990

FOREWARD

The contents of this report were developed under a grant from the Department of the Interior, U.S.Geological Survey. However, those contents do not necessarily represent the policy of the agency, and you should not assume endorsement by the Federal Goverment.

ABSTRACT

A comprehesive model has been developed for use in modeling the hydrologic response of rill network systems. The model, which is called HYMODRIN, is composed of both a hydrologic runoff component and a hydraulic channel routing component. The hydrologic component of the model uses a Green Ampt infiltration approach linked with a nonlinear reservoir runoff model. The channel routing component of the model is based on a finite element solution of the diffusion wave equations. In order to account for backwater effects the model employs a dual level iteration scheme.

The model may be used in either a stand alone mode or as part of a comprehensive integrated rill erosion model. In the latter case, the hydrologic data for the rill network and the associated interrill flow areas is provided by a geographic-hydrologic interface model called GHIM. This model accepts data from a digital elevation model and translates it into a form compatible with the hydrologic model.

This report contains the theoretical development and operating instructions for both GHIM and HYMODRIN. Computer listings for both programs are provided.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

HYDROLOGIC MODEL FOR DYNAMIC RILL NETWORKS

I. INTRODUCTION

Soil erosion continues to be a major contributor to environmental pollution. Sediment yield from disturbed lands can be as much as 10,000 times that from comparable undisturbed lands, causing serious problems in our nations's waterways. Chemicals adsorbed on the exchange phase of the collodial sediment particles add to the pollution problem. Topsoil loss resulting from erosion represents a serious drain on our natural resources.

The effectiveness of erosion control planning will be dependent upon the level of sophistication of the planning effort. Extensive data collection in the field can be done, but this is time consuming and expensive. Another option is to use mathematical models capable of simulating erosion.

Recent studies at the University of Kentucky have shown that the microrelief of a watershed can have a significant impact on the resulting erosion rates. Unfortunately, the majority of existing watershed sedimentology models fail to consider microrelief and rill network development. Instead, such models rely on generalized functional relationships whose parameter values have been synthesized from extensive field tests. As a result, these parameter values tend to reflect average conditions which may or may not be present in the actual watershed.

Recently, efforts have been made to develop a model of rill and interrill erosion in which predictions are made for each individual rill rather than gross estimates for a watershed. In order to make such predictions, it is necessary to describe the following phenomena on the watershed: (1) The location and growth of each rill, (2) The flow into and through each rill, and (3) The sediment movement into each rill and within each rill. In the current research effort these processes are modeled using a comprehensive modeling structure which integrates the separate processes into a single modeling environment.

As a result of an ongoing research effort at the University of Kentucky, a new comprehensive hydrology and sedimentology model has been developed for dynamic rill networks. The new model has the capability to identify the composition of rill networks in a watershed as well as predict the erosion and transport of sediment from the watershed in response to a specified times series of rainfall events. The model, which is called DRM5 (Dynamic Rill Model 5), is composed of five main components: a random surface generator (RSG), a digital elevation model (DEM), a geographic hydrologic interface model (GHIM), a hydrologic model (HYMODRIN), and an erosion model (DERM). For a given hydrologic response area the generation of the rill network and resulting erosion process is modeled through a serial application of each of the five subprograms (see Figure 1.1).

Starting with an initial charaterization of the soil matrix and microrelief structure the development model is used to generate an initial network of preferential flow paths. The microrelief structure and preferential rill pattern are then passed to the hydrologic model which uses this information along with additional hydrologic inputs (i.e. rainfall, soil moisture storage, etc.) to generate flow depths and velocities over the entire response area. These values are then passed to the sedimentology model where they are used to determine the amount of eroded soil as well as the change to the microrelief structure of the area. Potentially, the changes to the microrelief structure could then be passed back to the rill growth model where they could be used to define a new modified rill pattern which could then passed to the hydrologic model for analysis of the next rainfall event. This cyclic process could be repeated for each rainfall event until an entire rainfall series is analyzed. The results of the model include a time series of runoff and sediment loads as well as a time-space history of the modeled response area.

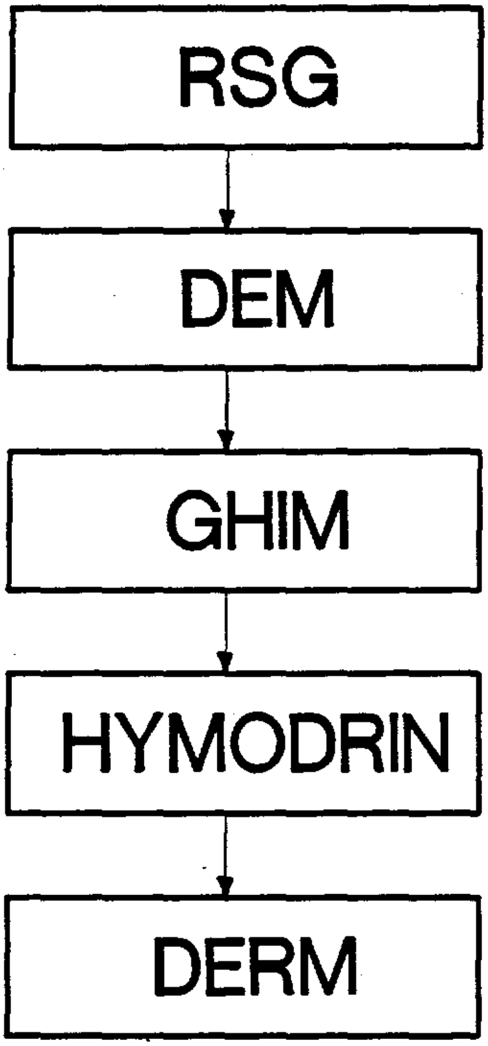

Figure 1.1. Schematic of DRM3

II. GEOGRAPHIC-HYDROLOGIC INTERFACE MODEL

The purpose of this chapter is to describe the geographic-hydrologic interface model (GHIM) which links the rill development model with the hydrologic model. The interface model is composed of an executive block which reads the input data and makes subroutine calls to two subroutines: SRILL and IRILL. The subroutine SRILL generates the hydrologic characteristics of each subrill while IRILL generates the hydrologic characteristics of each interrill flow element. Before describing the various components of the model a brief explanation of the required rillshed characterization is provided.

2.1. RILLSHED CHARACTERIZATION

In using HYDROMIN, an explicit characterization of the rillshed must be provided. The program assumes that a hillslope or field may be divided into a matrix of response cells (see Figure 2.1). The resulting cells may then be grouped into distinct subsets called either subrillsheds or null-rillsheds. Subrillsheds are composed of cells which drain to a common subrill. Null-rillsheds are composed cells which do not drain to a common subrill but drain directly off the drainage area. Distinct collections of subrillsheds which drain to a common point are called rillsheds. A given hillslope or field may thus be characterized as a collection of rillsheds and null-rillsheds.

As shown in Figure 2.1., each rill network may be characterized by a number of subrills, each of which drains an associated subrillshed. Each single rill or individual subrill may be further subdivided into individual subrill elements. Each individual subrill element may accept runoff from up to two adjacent interrill flow elements and from an additional upstream element (either another subrill element or an upstream source element). Each of these various components is illustrated in Figure 2.1.

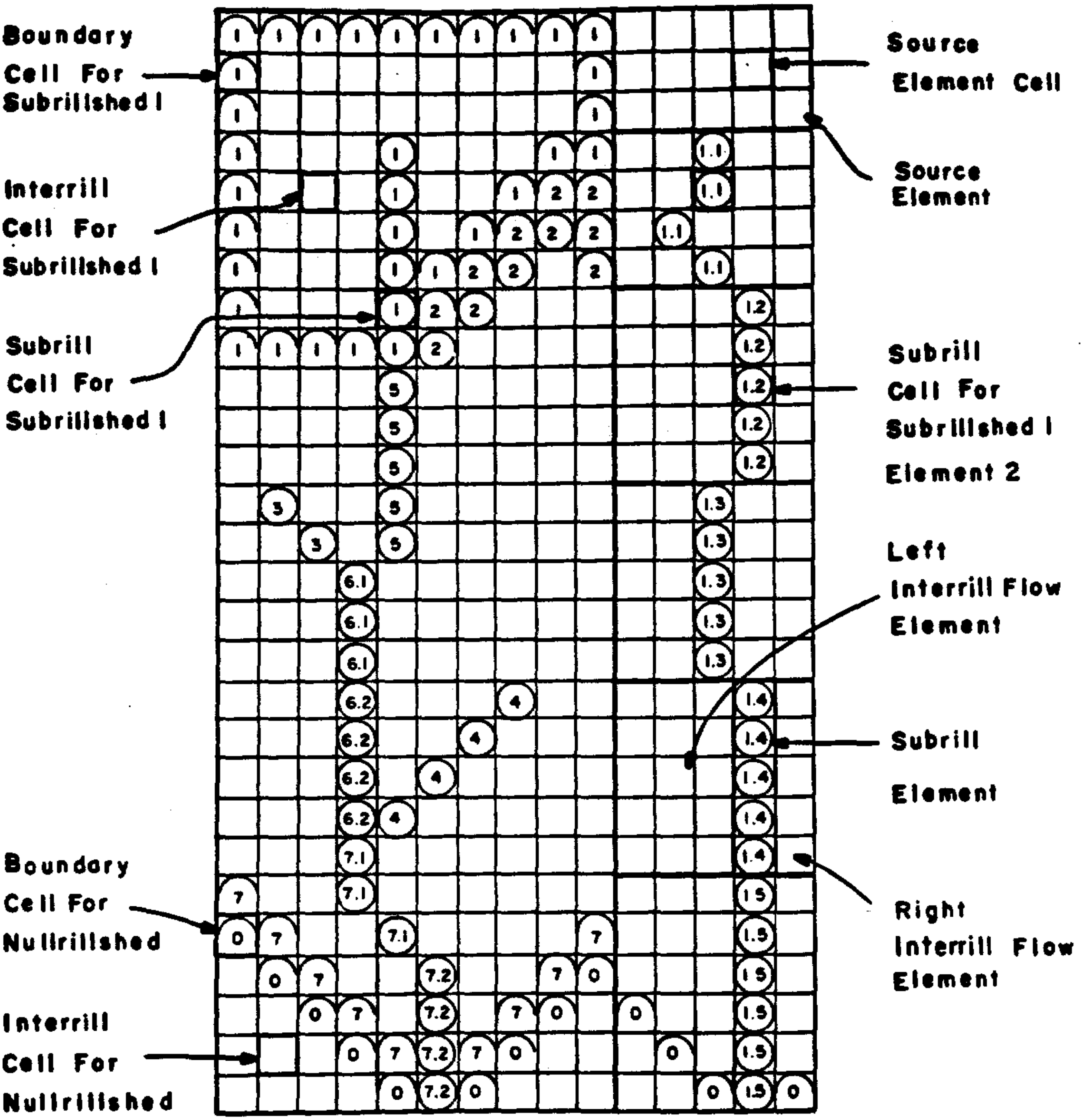

Figure 2.1. Rillshed Characterization

Note: A period in a subrillshed denotes elements, i.e., 6.1, 6.2, etc. If subrillshed 6 had only one element, it would be denoted as 6 without a period and a suffix.

2.2. GEOGRAPHIC DATA FILES

The geographic-hydrologic interface program requires four different input data files in order to generate the network characteristics required by HYDROMIN. Each file is generated by the rill development model (RDM). These files include: NDATA, SDATA, IDATA, and EDATA. Using these four files GHIM creates a single output file (i.e., HDATA) for use by HYDROMIN (see Figure 2.2). Detailed descriptions of the formats of each file are provided in Appendix A.

The NDATA data file contains descriptive information on how the rill network is arranged along with the total number of subrill cells and interrill cells associated with each subrillshed. The SDATA file contains the X and Y coordinates of the cells associated with each subrill identified in the NDATA file. This file also contains the cummulative number of upstream cells associated with each subrill cell. The IDATA file contains the X and Y coordinates of the interrill cells associated with each subrillshed and each nullrillshed identified in the NDATA file. Finally, the EDATA file contains the elevation of each cell in the hydrologic response grid.

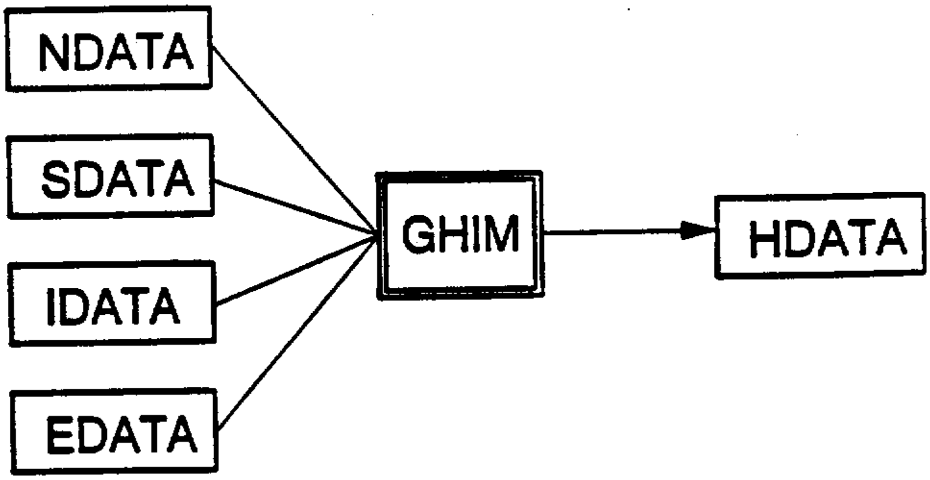

Figure 2.2 Geographic Data Files

2.3. HYDROLOGIC PARAMETERS

The geographic-hydrologic interface model (GHIM) reads each of the previously discussed input files, processes the data, and then creates a single output file for use with HYDROMIN. The interface program uses information from the input files to generate area, length, slope and roughness of each flow element. The algorithms used to generate these values from the input geographic information are discussed in the following sections.

2.3.1. Interrill Flow Area

Each subrillshed may be subdivided into several pairs of interrill flow elements, with one element on both sides of each subrill element. The total number of interrill flow elements can be obtained by first specifying the number of subrill cells to be associated with each interrill flow element. Once this value has been set, the total number of interrill flow elements for a particular subrillshed may be obtained from the following equation:

$$NIRE_{ij} = 2 * \left[\frac{TSRC_{ij}}{NCPE} \right] \tag{2.1}$$

where $NIRE_{ij}$ = the number of interrill elements associated with rill i and subrill j, $TSRC_{ij}$ = the total number of subrill cells associated with rill i and subrill j, and $NCPE$ = the number of subrill cells per subrill element. The actual interrill cells associated with each interrill element can be identified by considering the cells in the rows bounded between the first and last subrill cells associated with the corresponding subrill element. The interrill cells associated with the left element can be identified by only considering those cells left of the subrill element while the interrill cells associated with the right element can be identified by only considering those cells right of the subrill element. The flow area for each interrill flow element is then obtained by multiplying the number of cells contained in the element by the unit area of each cell.

2.3.2. Element Flow Length

The length of each interrill flow element is obtained by averaging the lengths of each horizontal row of cells contained in the flow element (see Figure 2.3). Mathematically this may be expressed as:

$$L_{ijkl} = \frac{1}{N_k} \sum_{n=1}^{N_k} L_{ln} \qquad \text{for } l = 1,2 \tag{2.2}$$

where i = the rill index, j = the subrill index, k = the subrill element index, l = the interrill element index (1 = left element, 2 = right element), and N_k = the number of rows associated with subrill element k.

The length of each source flow element is obtained by averaging the lengths of each vertical column of cells contained in the flow element (see Figure 2.4). Mathematically this may be expressed as:

$$L_{ij} = \frac{1}{N_j} \sum_{n=1}^{N_j} L_n \tag{2.3}$$

where N_j = the number of columns associated with source element j.

2.3.3. Element Flow Slope

The slope of each interrill flow element is obtained by averaging the slope of each horizontal row of cells contained in the flow element. Mathematically, this may be expressed as:

$$S_{ijkl} = \frac{1}{N_k} \sum_{n=1}^{N_k} \frac{EIRBC_{ln} - ESRC_{ln}}{L_{ln}} \qquad \text{for } l = 1,2 \tag{2.4}$$

where $EIRBC_n$ = the elevation of the interrill boundary cell associated with row n, and $ESRC_n$ = the elevation of the subrill cell associated with row n.

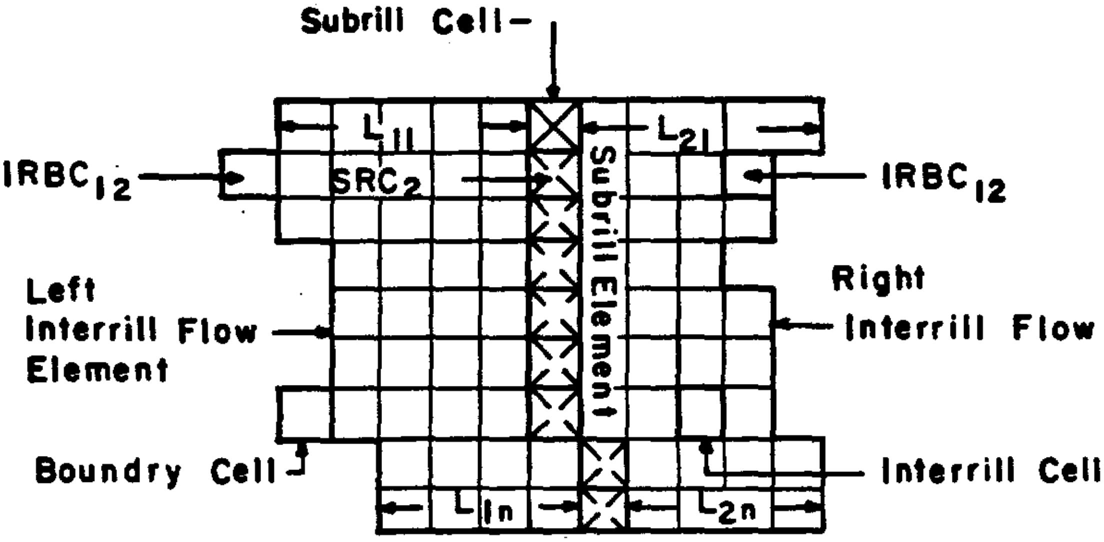

Figure 2.3 Interrill Element Characteristics

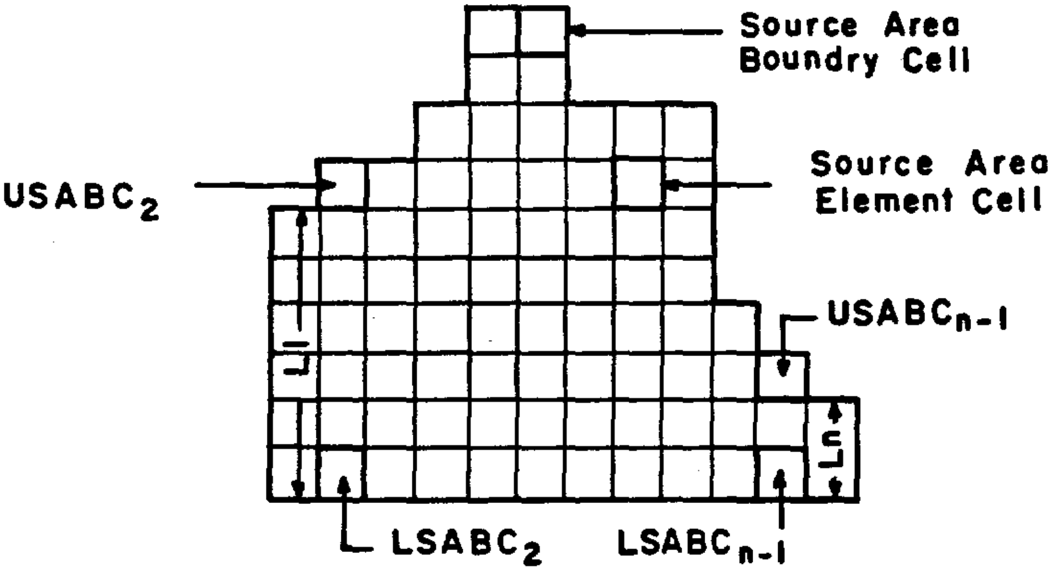

Figure 2.4 Source Element Characteristics

The slope of each source flow element is obtained by averaging the slopes of each vertical row of cells contained in the flow element. Mathematically, this may be expressed as:

$$S_{ij} = \frac{1}{N_j} \sum_{n=1}^{N_j} \frac{EUSABC_n - ELSABC_n}{L_n} \tag{2.5}$$

where $EUSABC_n$ = the elevation of the upper source area boundary cell associated with column n, and $ELSABC_n$ = the elevation of the lower source area boundary cell associated with column n.

2.3.4. Initial Random Roughness

The initial random roughness of each flow element is assumed to be equal to the standard error associated with the plane fit through the elevations associated with the cells in each interrill flow element. Mathematically, the random roughness may be expressed as:

$$RR_{ijkl} = \sqrt{\frac{SR}{N_{ijkl} - 3}} \tag{2.6}$$

where N_{ijkl} = the number of cells associated with flow element $ijkl$ (where k and $l = 0$ for each source element) and SR = the standard error which may be expressed as:

$$SR = \sum_{n=1}^{N_{ijkl}} (E_n - a_o - a_1{}^*X_n - a_2{}^*Y_n)^2 \tag{2.7}$$

where N_{ijkl} = the total number of interrill cells associated with flow element $ijkl$ (where k and $l = 0$ for each source element), E_n = the elevation associated with cell n, X_n = the X coordinate associated with cell n, Y_n = the Y coordinate assoicated with cell n, and a_o, a_1 and a_2 are the coefficients of the best fit plane. These coefficients may be obtained by solving the following set of normal equations.

$$
\begin{aligned}
&na_o + \sum X_i a_1 + \sum Y_i a_2 = \sum E_i \\
&\sum X_i a_o + \sum X_i^2 a_1 + \sum X_i Y_i a_2 = \sum X_i E_i \\
&\sum Y_i a_o + \sum X_i Y_i a_1 + \sum Y_i^2 a_2 = \sum Y_i E_i
\end{aligned}
\tag{2.8}
$$

2.4. PROGRAM EXECUTION

When executing GHIM, the user must specify the total number of subrill data sets contained in the NDATA, SDATA, and IDATA files as well as the maximum number of rows and columns associated with the EDATA set. The user must also specify the number of cells per element (i.e. NCPE) which is used in Eq. (2.1) to determine the total number of elements associated with each subrill. Detailed instructions for use of the geographic-hydrologic interface model (GIHM) are provided in Appendix A.

III. DETERMINISTIC HYDROLOGIC MODEL

The purpose of this chapter is to describe the deterministic hydrologic model. The deterministic hydrologic model is called HYMODRIN (HYdrologic Model Of Dynamic RIll Networks) and can be used in the DRM3 program environment or it can be applied as a stand alone model. The deterministic model is composed of an executive block which reads in the input data and makes subroutine calls to five different subroutines (see Figure 3.1). These subroutines include: RAINFALL, EXCESS, RUNOFF, RILL, ROUTE. Each of these subroutines are discussed in the following sections.

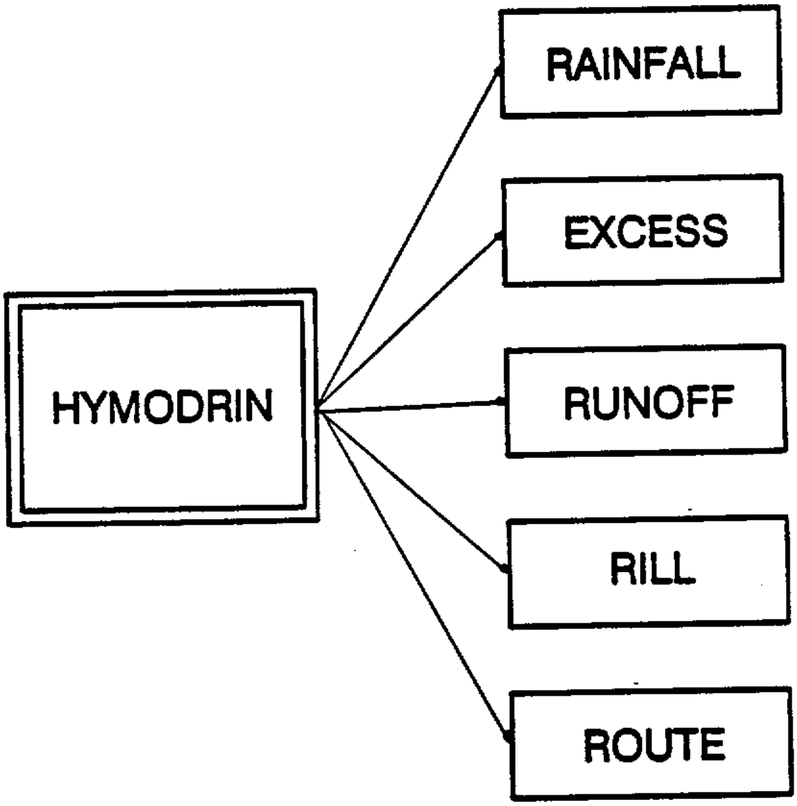

Figure 3.1. Flowchart of HYDROMIN

3.1. RAINFALL SUBROUTINE

The RAINFALL subroutine reads in the rainfall data and disaggregates the values into a time interval compatible with the computation interval of the main program. This subroutine also determines the kinetic energy associated with the rainfall event.

3.2.1. Rainfall Kinetic Energy

The cummulative kinetic energy for a rainfall event is determined in RAINFALL. The cummulative kinetic energy is used in the program to determine the decrease in depression storage over time. The cummulative kinetic energy CKE_t $(joules/cm^{2})$ at time t for a storm of duration T may be determined as follows (Wischmeier and Smith, 1958).

$$CKE_t = .0001 \sum_{i=1}^{t} (11.9 + 8.73 * Log(r_i)) * r_i * \delta_t \tag{3.1}$$

where r_i = the rainfall intensity for time step i, (mm/hr), and δt = hours.

3.2. EXCESS SUBROUTINE

The EXCESS subroutine determines the rainfall excess vector for the total rill system. Infiltration is computed using a dual layer Green Ampt model.

3.2.1. The Green Ampt Equation

In using the Green Ampt (1911) equation the infiltration process is conceptualized as a plug flow process in which a saturated plug moves down vertically through the soil matrix (see Figure 3.2). The Green Ampt equation assumes that the infiltration rate f (cm/hr) can be expressed as a function of the hydraulic conductivity K (cm/hr), the depth to the wetting front L (cm), and the average capillary suction across the wetting front S (cm). This relationship may be expressed as:

$$f = \frac{K\,(\,L + S)}{L} \tag{3.2}$$

Noting that the cummulative infiltration, F (cm), may be expressed as:

$$F = (\theta_s - \theta_i\,)\,L \tag{3.3}$$

where θ_s is the saturated moisture content and θ_i is the intial moisture content, the Green Ampt equation can now be expressed as:

$$f = K\,(1 + \frac{IMD^*S}{F}) \tag{3.4}$$

where IMD is the initial moisture deficit (fraction) which is the difference between the saturated moisture content and the initial moisture content. Integration of Eq. (3.4) yields the cummulative form of the Green Ampt equation:

$$F - S^*IMD^*ln(1 + \frac{F}{S^*IMD}) = K^*t \tag{3.5}$$

where t = cummulative time in hours. This equation is nonlinear in terms of F. It may be solved by the method of successive substitution by rearranging it to read:

$$F = K^*t + S^*IMD^*ln(1 + \frac{F}{S^*IMD}) \tag{3.6}$$

Given K, t, S, and IMD, a trial value of F is substituted on the right hand side and a new value of F is calculated on the left hand side. This new value may be substituted on the right hand side and the process repeated until the calculated values of F converge to a constant. A good value to use for an intial estimate of F is K^*t (Chow, et. al., 1988).

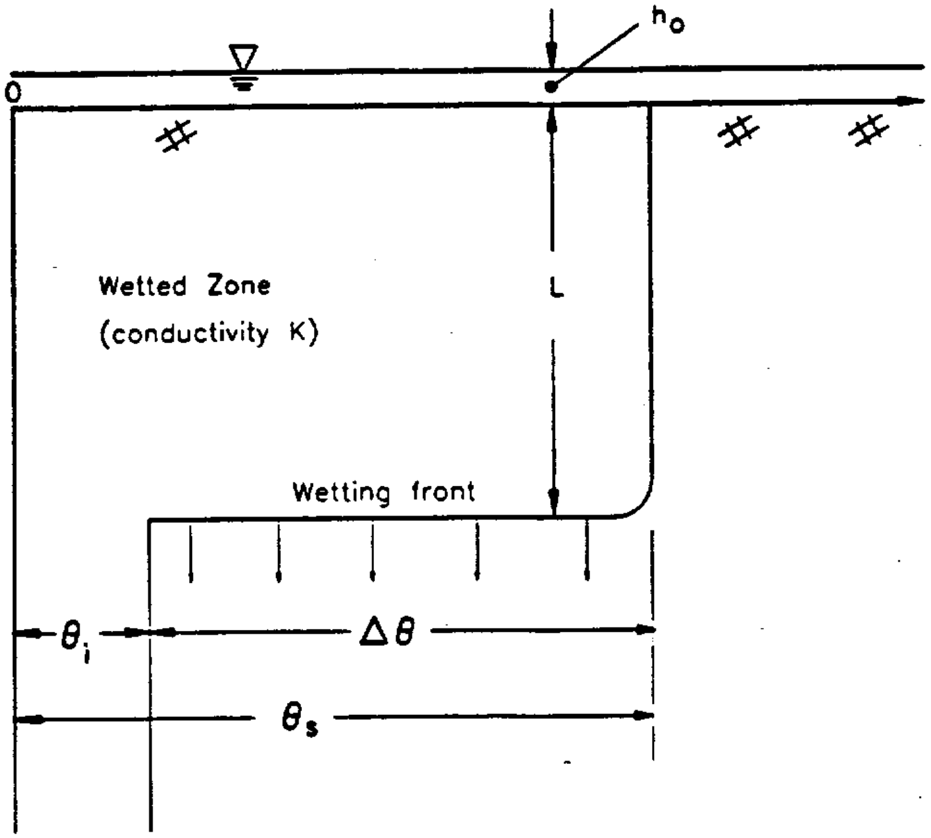

Figure 3.2 Green Ampt Infiltration Model

3.2.2. Two Layer Green Ampt Equation

The infiltration model used in HYDROMIN is based on a two layer application of the Green Ampt equation (Chow et. al., 1988), and may be conceptualized as shown in Figure 3.3. In this model the soil matrix is divided into two layers. The upper layer has Green-Ampt parameters of K_1, S_1, and IMD_1 while the lower layer has Green-Ampt parameter of K_2, S_2, and IMD_2. As long as the wetting front is above the second layer (i.e. $L < F/IMD_1$) the infiltration rate is given by Eq. (3.4) and the cummulative infiltration volume may be determined by Eq. (3.6). If the wetting front extends beyond the upper layer (i.e. $L > F/IMD_1$) then the infiltration rate is given by:

$$f = \frac{K_1\,K_2}{H_1K_2 + L_2K_1}\,(S_2 + H_1 + L_2) \quad , \tag{3.7}$$

and the cummulative infiltration is given by:

$$F = H_1 IMD_1 + L_2 IMD_2 \tag{3.8}$$

By combining Eqs. (3.7) and (3.8) into a differential equation for L_2 and integrating one obtains:

$$L_2\,\frac{IMD_2}{K_2} + \frac{1}{K_1K_2}\left[IMD_2H_1K_2 - IMD_2K_1(S_2 + H_1)\right] ln(1 + \frac{L_2}{S_2 + H_1}) = t \tag{3.9}$$

As with Eq. (3.6) this equation may be solved in terms of L_2 using successive substitution. Once the value of L_2 has been obtained it can be used to determine both f and F using Eqs (3.7) and (3.8).

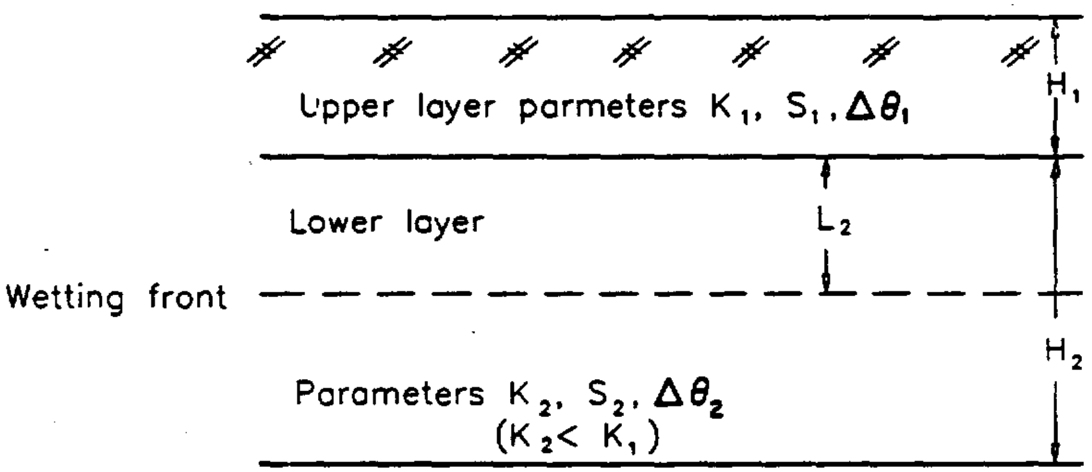

Figure 3.3 Two Layer Green Ampt Model

3.2.3. Boundary Conditions

Equations (3.4) and (3.5) are only valid if water is ponded on the surface of the soil. For those instances where the soil surface is not initially saturated the equations must be modified to account for the time it takes for the soil to become saturated, which is known as the ponding time. For the case of a constant rainfall intensity r (cm/hr), the ponding time may be expressed as:

$$t_p = \frac{K * S * IMD}{r^*(r - K)} \tag{3.10}$$

The volume of water that must be infiltrated before the soil surface becomes saturated is known as the ponding volume. This volume may be expressed as:

$$F_p = r^*tp \tag{3.11}$$

Both Eqs. (3.10) and (3.11) are based on the assumption that all rainfall infiltrates the soil prior to the ponding time. In order to reflect the impact of antecedent ponding volume on the Green-Ampt infiltration equation, the time axis of the infiltration curve must be shifted to a new time origin t_o. The value of the new time origin may be obtained using the following equation:

$$t_o = t_p - \frac{1}{K}\left[F_p - S^*IMD^*ln(1 + \frac{F_p}{S^*IMD}) \right] \tag{3.12}$$

3.2.4. Computation Methodology

The computation steps required to evaluate the two layer Green-Ampt equation may be summarized as follows:

0) Determine t_p, F_p, and t_o for given values of K_1, S_1, IMD_1, and r.

1) Determine the initial incremental infiltration volume ΔF where
   $\Delta F = r\Delta t$.

2) Determine the cummulative infiltration volume for time step t where
   $F_t = F_{t-1} + \Delta F$.

3) If $F_t < F_p$ set $f = r$ and go to 9), otherwise continue.

4) Determine the infiltration volume in the first layer (i.e. F_1) using
   Eq. (3.6) with $t = t - t_o$.

5) If the wetting front is in the upper layer (i.e. $H_1 > F/IMD_1$) then determine f using
   Eq. (3.4) (where $t = t - t_o$ and go to 9) otherwise continue.

6) Determine the depth of the wetting front in the second layer (i.e. L_2)
   using Eq. (3.9) where $t = t - t_o$.

7) Determine the cummulative infiltration using Eq. (3.8).

8) Determine the current infiltration rate using Eq. (3.7).

9) Determine the rainfall excess intensity e, where $e = r - f$.

10) Update the time step and return to step 1.

3.2.5. Infiltration Parameters

Application of the Green Ampt equation requires estimates of the hydraulic conductivity K, the initial soil moisture content IMD, and the wetting front suction head S. The initial soil moisture deficit can be expressed as:

$$IMD = (\theta_s - \theta_i) \tag{3.13}$$

where θ_s is the staturated moisture content and θ_i is the initial moisture content. While θ_i will be a function of the antecedent rainfall conditions, the remaining required parameters (i.e. K, θ_s, and S) are a function of the soil characteristics. Estimates of these parameters may be obtained using Table 3.1 (Rawls, et. al., 1983).

Table 3.1 Green Ampt Infiltration Parameters

Soil class	Saturated Moisture Content θ_s	Wetting Front Soil Suction head (cm) S	Hydraulic Conductivity (cm/hr) K
Sand	0.437 (0.374-0.500)	4.95 (0.97-25.36)	11.78
Loamy sand	0.437 (0.363-0.506)	6.13 (1.35-27.94)	2.99
Sandy loam	0.453 (0.351-0.555)	11.01 (2.67-45.47)	1.09
Loam	0.463 (0.375-0.551)	8.89 (1.33-59.38)	0.34
Silt loam	0.501 (0.420-0.582)	16.68 (2.92-95.39)	0.65
Sandy clay loam	0.398 (0.332-0.464)	21.85 (4.42-108.0)	0.15
Clay loam	0.464 (0.409-0.519)	20.88 (4.79-91.10)	0.10
Silty clay loam	0.471 (0.418-0.524)	27.30 (5.67-131.50)	0.10
Sandy clay	0.430 (0.370-0.490)	23.90 (4.08-140.2)	0.06
Silty clay	0.479 (0.425-0.533)	29.22 (6.13-139.4)	0.05
Clay	0.475 (0.427-0.523)	31.63 (6.39-156.5)	0.03

Table ranges in parenthesis

3.3. RUNOFF SUBROUTINE

The RUNOFF subroutine determines the runoff from each interrill flow element using a nonlinear reservoir approach. This subroutine also calculates the depression storage and the Chezy roughness coefficient for each flow element. In HYDROMIN, the depression storage is allowed to vary with time as a function of changes in the random roughness of the plane. The random roughness of each plane will decrease over time as a function of the cummulative kinetic energy applied to the plane.

3.3.1. Nonlinear Reservoir Model

All interrill flow hydrographs are generated using a nonlinear reservoir approach. The flow equations for the nonlinear reservoir are obtained by combining Chezy's equation with a finite difference representation of the continuity equation to yield the following nonlinear relationship (Huber et. al., 1981):

$$F(\delta d) = \delta d - \delta t \ (\alpha_e \ d^{3/2} + e) \tag{3.14}$$

where $\delta d = d_2 - d_1$ (*ft*); d_2 = the interrill flow depth at time step 2 (*ft*); d_1 = the interrill flow depth at time step 1 (*ft*); δt = the time interval (*sec*); α_e = overland flow element constant; $d = d_1 - d_s + \frac{\delta d}{2}$ (where d_s = the depression storage); and e = the rainfall excess intensity during time interval δt (*ft/sec*). The interrill flow element constant α_e may be expressed as follows:

$$\alpha_e = \frac{-(C^*W^*S^{3/2})}{A_e} \tag{3.14}$$

where C = Chezy's roughness coefficient; W = the width of the interrill flow element (*ft*); S = the slope of the interrill flow element (*ft/ft*); and A_e = the surface area of the interrill flow element (*ft*^{2}). Eq. (3.15) may be solved in terms of δd for each time step using Newton's method. Once δd has been determined, the corresponding discharge may be obtained from Chezy's equation.

3.3.2. Depression Storage

The subroutine STORE utilizes the interrill flow element slope and initial random roughness to estimate the initial available depression storage using curves developed by Linden (1979). The representation of these curves for use in HYMODRIN is based on a four point Lagrangian approximation developed by Hirshi (1985). A plot of the equations are shown in Figure 3.4.

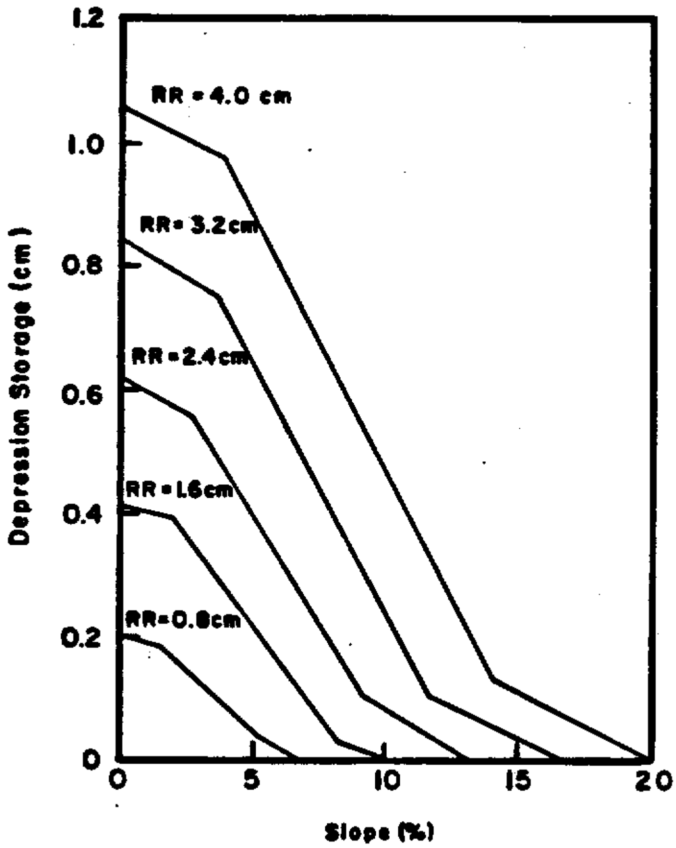

Figure 3.4 Roughness Curves

3.3.3. Random Roughness

The element depression storage is a function of the random roughness of the element. As the random roughness changes over time so will the associated value of the depression storage element for a particular time step can be expressed as a function of the intital random roughness RR_o and the cummulative kinetic energy CKE_t $(joules/cm^2)$ as follows (Moore, 1979):

$$RR_t = RR_o\ e^{-2*CKE_t} \tag{3.16}$$

3.3.4. Chezy Roughness Coefficient

The Chezy roughness coefficient for the interrill response elements can be obtained using the following relationship (Gilley et. al., 1989):

$$C = [\frac{8g}{f}]^{1/2} \tag{3.17}$$

where:

$$f = 4.0*[\frac{3.42^{clay}}{12.42^{sand}}] \tag{3.18}$$

and where g = the gravitational constant, *sand* = the fraction of sand, and *clay* = the fraction of clay.

3.4. RILL SUBROUTINE

The RILL subroutine generates the required data for each subrill element. This includes the finite element data required by ROUTE for use in routing the resulting hydrographs, as well as the width of each subrill.

3.4.1. Determination of the Subrill Width

The width of each subrill in each rill network is determined using the equilibrium width model of Lane and Foster (1980). This model assumes that the geometry of each rill may be approximated with a rectangular cross section. Using this model, the width of each rill may be expressed as follows:

$$W_r = \left(\frac{qn}{S^{1/2}}\right)^{3/8} \frac{W_*}{R_*^{5/8}} \tag{3.19}$$

where W_r = the width of the rill (m), q = the peak discharge in the rill (m^3/s), n = the Mannings roughness coefficient, S = the rill slope (m/m), W_* = the normalized rill width, and R_* = the normalized hydraulic radius. Both the normalized width (W_*) and the normalized hydraulic radius (R_*) may be expressed as a function of the rill conveyance parameter x_{*_c} as shown in Figures 3.5 and 3.6. The rill conveyance parameter x_{*_c} can likewise be expressed as a function of the rill conveyance function as shown in Figure 3.7. The rill conveyance function can be expressed as a function of the physical characteristics of the rill as follows:

$$g(x_{*_c}) = \left(\frac{qn}{S^{1/2}}\right)^{3/8} \frac{\gamma S}{\tau_c} \tag{3.20}$$

where τ_c = the critical tractive force of the soil (N/m^2), and γ = the specific weight of water (N/m^3).

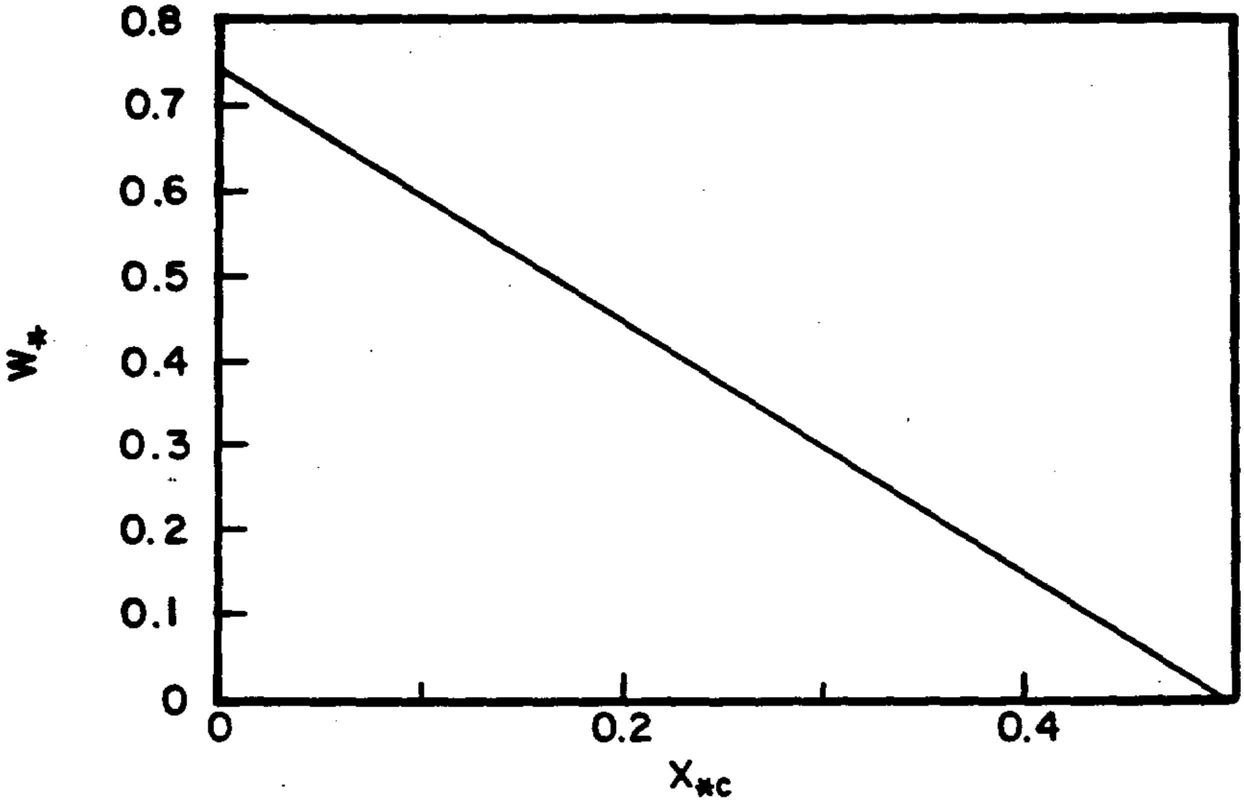

Figure 3.5 Normalized Width Function

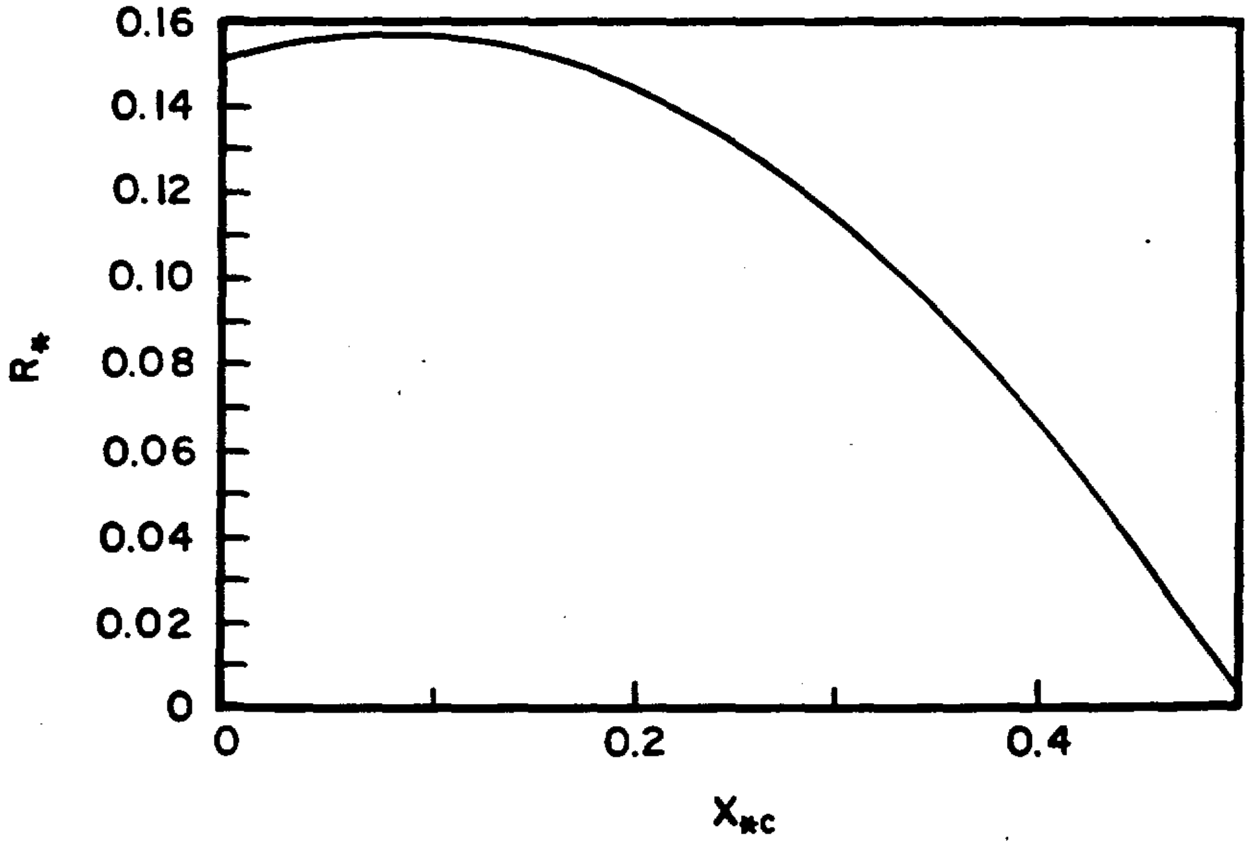

Figure 3.6 Normalized Hydraulic Radius Function

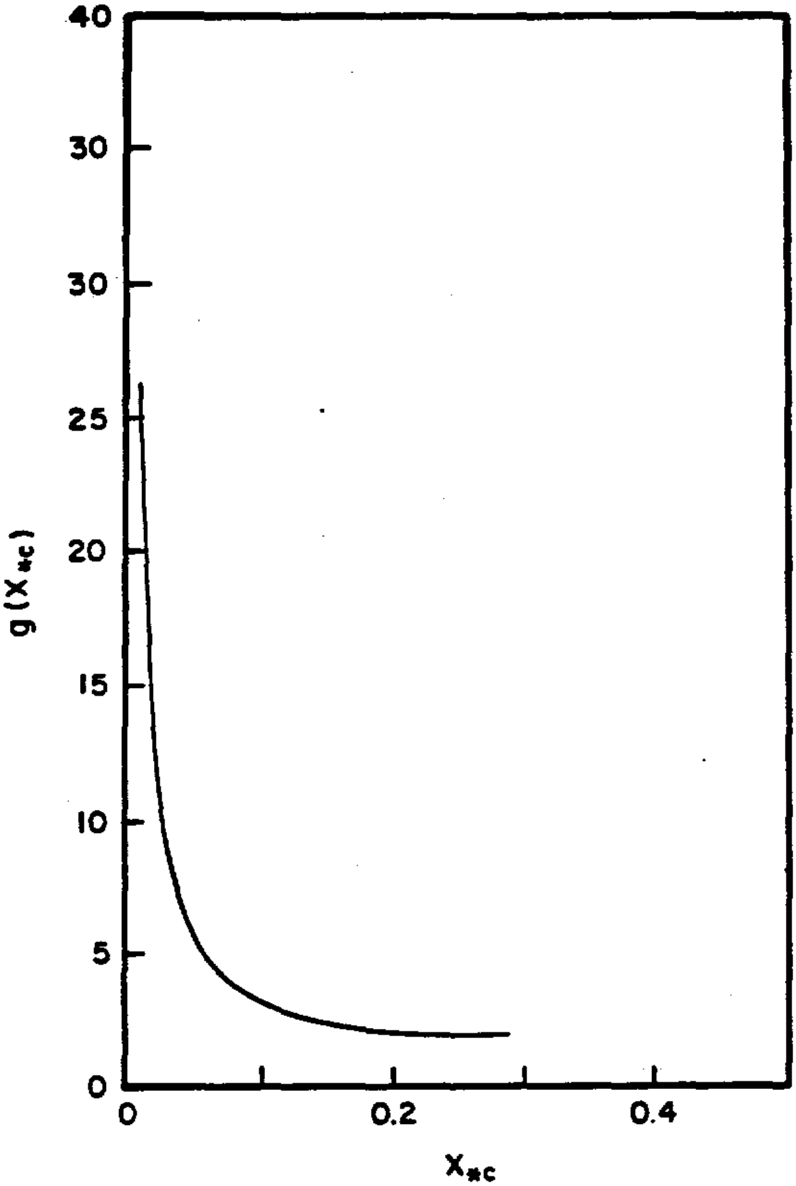

Figure 3.7 Rill Conveyance Function

3.4.2. Computation Methodology

The width of each subrill may now be calculated as a function of the flow and subrill parameters using the following steps:

1) Calculate the conveyance function $g(x_{*_c})$ using Eq. (3.20).

2) Given $g(x_{*_c})$, obtain a value for x_{*_c} from Figure 3.7.

3) Given a value for x_{*_c}, obtain values for W_* and R_* from Figures 3.5 and 3.6.

4) Calculate the subrill width using Eq. (3.19).

3.4.3. Initial Conditions

In order to determine the width of each subrill, the peak discharge must be specified. The peak discharge in a particular subrill is not known until after each interrill runoff hydrograph has been routed through the rill network. However, in order to route the interrill runoff hydrographs through the rill network system the width of each subrill must be known. This problem is addressed in HYDROMIN by first approximating the peak discharge in each rill by using the rational equation. In this context the peak discharge in each rill is obtained as follows:

$$q_r = avg_r A_r \tag{3.21}$$

where avg_r = the maximum rainfall excess intensity and A_r = the total drainage area upstream of subrill r. These discharges are then used to obtain initial values of the widths for each subrill. These widths are then used in ROUTE to generate the hydrographs in each subrill. The peak values associated with each hydrograph can then be used to update the channel widths for subsequent applications of the program.

3.5. ROUTE SUBROUTINE

The ROUTE subroutine accepts the interrill flow element hydrographs generated by RUNOFF and then routes them through the associated rill network system. The hydrographs are routed through each subrill using a finite element formulation of the diffusion wave equations. The following discussion of the general hydraulic equations and associated solution methodologies is taken from Fread (1982).

3.5.1. Saint Venant Equations:

The general equations governing unsteady flow in an open channel are known as the Saint Venant (1871) equations. These equations consist of a conservation of mass equation:

$$\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} = \int_{\sigma} q \; d\sigma \tag{3.22}$$

and a conservation of momentum equation:

$$\frac{1}{gA}\frac{\partial Q}{\partial t} + \frac{1}{gA}\frac{\partial}{\partial x}\left(\frac{\beta}{A}Q^2\right) + \frac{\partial}{\partial x}(kY) + (k-k')\,\frac{Y}{A}\frac{\partial A}{\partial x}$$

$$= S_o - S_f + \frac{1}{\gamma A}\frac{\partial T}{\partial x} + \frac{1}{gA}\int_{\sigma} q \; U_x \; d\sigma \tag{3.23}$$

in which t = time, x = distance along the channel, Y = depth of flow measured vertically, A = channel cross sectional area, g = gravitational acceleration, Q = channel flow, q = lateral inflow, β = momentum flux correction factor, S_o = the channel slope, S_f = the friction slope, γ is the specific weight of the fluid, U_x = the x-component velocity of the lateral flow when joining the channel flow, and k, and k', are pressure distribution correction factors. Normally, when dealing with gradually varied flows the pressure distribution correction factors and the momentum flux correction factor are assumed to be equal to one and the variation of T with respect to x is neglected (Yen, 1973).

When applying the Saint Venant equations to simulate unsteady flow in a channel, the friction slope, S_f, is usually estimated using Manning's equation:

$$S_f = \left(\frac{n^2 V^2}{C_n}\right) R^{-4/3} = \left(\frac{n^2 Q^2}{C_n A^2}\right) R^{-4/3} \tag{3.24}$$

in which n is Manning's roughness factor, R is the hydraulic radius, which is equal to A divided by the wetted perimeter, and $C_n = 1$ for SI units and 2.22 for English units. The friction slope may also be estimated by using the Darcy Weisbach formula:

$$S_f = \frac{f}{8gR} V^2 = \frac{f}{8gR} \frac{Q^2}{A^2} \tag{3.25}$$

in which f is Weisbach's resistance coefficient; or by the Chezy formula:

$$S_f = \frac{V^2}{C^2 R} = \frac{Q^2}{C^2 R A^2} \tag{3.26}$$

in which C is Chezy coefficient. Rigorously speaking, the values of n, f, and C for unsteady nonuniform flow have not been established. They are functions of flow unsteadiness, nonuniformity, the Reynold's and Froude numbers, and the channel boundary roughness conditions including bed forms for alluvial channels (Yen, 1973).

3.5.2. Kinematic Wave Equations:

Due to the complexities of the Saint Venant equations, various simplified approximations of flood wave propagation have been developed. The most common simplification is the kinematic wave approximation. The concept of the kinematic wave implies that inertia forces are negligible relative to gravitational and frictional forces, and that flow is a function of depth alone. This results in a numerical solution of the continuity equation and an analytical solution of the momentum equation. The kinematic model uses the following simplified form of the conservation of momentum equation:

$$S_o = S_f \tag{3.27}$$

Equation (3.27) essentially states that the momentum of the unsteady flow is assumed to be the same as that of steady uniform flow as described by the Chezy or Manning equation or some other similar expression in which discharge is a single-valued function of stage, i.e.,

$$Q = \alpha y^m \tag{3.28}$$

Combining Eq. (3.22) and Eq. (3.28) results in the following non-linear kinematic wave model (Li, et. al., 1975):

$$\frac{\partial A}{\partial t} + \alpha m y^{m-1} \frac{\partial y}{\partial x} = \int_{\sigma} q \, d\sigma \tag{3.29}$$

which can be solved using either finite element or finite difference methods.

Lighthill and Whitham (1955), in a theoretical study of kinematic flow, set forth the justification for its application and cleared the way for its acceptance as an approach to flood routing. In doing so, they established that, for flow to be classified as kinematic, the Froude number should be less than two. An additional criterion was developed by Woolhiser and Ligget (1967) who utilized a dimensionless kinematic flow number, K, defined as:

$$K = \frac{L\,S\,g}{V^2} \tag{3.30}$$

When K is greater than 10, the dynamic wave is small and the kinematic wave solution approximates the solution obtained by the complete equations. They were further able to determine that an error of approximately 10% results when $K = 10$ and that an increase in the value of K indicates a rapid decrease in the magnitude of the error.

Eagleson (1970) suggested that both of the above conditions be upheld for use of the kinematic wave approximation in unsteady flow simulations; however, Al-Mashidani and Taylor (1974) have shown that the Froude number can be greater than two provided that the value of K is large.

3.5.3. Diffusion Wave Equations

As mentioned earlier, the full Saint Venant equations consist of the continuity equation given by Eq. (3.22) and the complete momentum equation depicted by Eq. (3.23). It has also been stated that for some situations, the inertial and pressure forces may be insignificant, so that, only the gravity force and the friction force terms need be retained. As shown previously, such an approach leads to a routing model known as the kinematic wave model given by Eqs. (3.22) and (3.28). However, there may be situations when, while the kinematic wave model may be grossly inadequate, the complete Saint Venant equations will be overly rigorous. In such cases, the inertial forces terms may be ignored while keeping the pressure force term in addition to the gravity force and the friction force terms. The resulting routing model is termed the diffusion wave model. The diffusion wave approximation uses the continuity equation defined by Eq. (3.22) and the following simplification of the momentum equation:

$$\frac{\partial}{\partial x}(kY) = S_o - S_f + \frac{1}{gA}\int_{\sigma} q\ U_x\, d\sigma \tag{3.31}$$

All the terms in Eq. (3.31) have been defined previously.

When applying the diffusion equations, the following general guidelines must be given due consideration. If the slope of the channel is steep and the flow is supercritical or near critical, the kinematic wave equations may be a reasonable approximation. However, for milder slopes, it may be more realistic to employ the diffusion wave equations in order to include the backwater effect or the propagation of waves upstream. For flat slopes and subcritical flow however, even the diffusion wave approach may not be adequate to simulate the significant backwater effects that may be present. In such cases, it may be essential to utilize the full Saint Venant equations.

Ponce et al. (1978) proposed the following criteria for the application of the kinematic-, diffusion-, or dynamic-wave (full St. Venant equations) model:

$$T_B S_o \left(\frac{v_o}{y_o}\right) \geq 171 \qquad \textit{Kinematic} \tag{3.32}$$

$$T_B S_o \left(\frac{g}{y_o}\right)^{1/2} \geq 30 \qquad \textit{Diffusion} \tag{3.33}$$

$$T_B S_o \left(\frac{g}{y_o}\right)^{1/2} < 30 \qquad \textit{Dynamic} \tag{3.34}$$

where T_B is the flood wave period, S_o is the channel bottom slope, v_o and y_o are the initial velocity and depth, and g is gravity acceleration. Thus, for shallow flow on steep slopes as well as for long duration flood waves, the kinematic wave approximation may be valid. Similarly, the diffusion wave scheme is useful for a greater variety of situations. The Saint-Venant equations, on the other hand, can be used for any condition.

3.5.4. Solution Techniques

Numerical methods for solving the dynamic wave equations gained popularity because the Saint Venant equations are partial differential equations which cannot be solved analytically except for a few simple situations. With the advent of high-speed digital computers, Stoker (1953) and Isaacson et al., (1954) first attempted to use the complete Saint Venant equations for flood routing on the Ohio river. Since then, much effort has been expended on the development of dynamic wave models, and the literature contains many dynamic models. These models can be categorized according to finite difference or finite element methods.

3.5.4.1. Finite Difference Methods

Finite difference methods may be classified as direct and characteristic methods. In direct methods, finite difference approximations are directly substituted into the governing partial differential equations given by Eqs. (3.22) and (3.23), and then incremental solutions are obtained for incremental times Δt and incremental distances Δx along the waterway. In the method of characteristics, the partial differential equations (3.22) and (3.23) are first converted into an equivalent set of four ordinary differential equations which are then approximated with finite differences to obtain solutions (Fread, 1985). Finite difference models can be classified further as either explicit or implicit, depending on the type of finite difference scheme that is used. Explicit schemes transform the differential equations into a set of easily manipulated algebraic equations which can be solved directly for each point on a time line. In contrast, implicit schemes convert the differential equations into a set of algebraic equations which must be solved simultaneously providing solutions for all points on a time line concurrently. The set of simultaneous equations may be either linear or nonlinear, the latter requiring an iterative solution procedure.

The important point to be noted in regard to the explicit and the implicit schemes is that the former, while being simpler to implement on a computer, can become numerically unstable if small time and space increments are not used. The latter (i.e., the implicit scheme), although substantially more complicated in terms of mathematical formulation, has been found to retain numerical stability for large time steps with little loss of accuracy. Hence, in general, implicit methods are considered to be more efficient and expeditious than the explicit method.

3.5.4.2. Finite Element Methods:

One of the more recent methods of solving the flow routing models is the finite element method. The method involves converting the integral form of the governing partial differential equations into a system of simultaneous algebraic equations by using a piecewise application of either the variational or the weighted residual principle on a finite number of elements which constitute the problem domain. The resulting system of equations are then solved to find the magnitude of the unknown variables.

The finite element formulation of any problem involves five basic components. These include the the governing differential equation, the elements, the nodes, the field variable, and the interpolating function. In applying the finite element method to a particular problem, the domain of the problem is divided into a finite number of subdomains called elements, which collectively approximate the shape of the physical continuum. In applying the finite element method to the channel routing problem, the channel is treated as a one-dimensional continuum. Nodes are located on element boundaries and sometimes within elements. For the channel flow problem the channel area is specified as the field variable. Values of the field variable are determined at each node for each time step. An interpolating function (usually a polynomial), defined in terms of the nodal values of the field variable, is used to approximate the flow area over each channel element. A different function is defined for each element; but the element functions are selected based on the continuity requirements of the governing equations, so that continuity is maintained between elements (Bickford, 1990).

The goal of any finite element formulation is to establish a system of algebraic equations which can be solved to obtain the nodal field variable values. The five basic steps for establishing the channel flow algebraic equations are: (1) construction of the integral relationships corresponding to the governing equation, (2) approximation of the field variable behavior, (3) establishment of the element equations, (4) time integration, and (5) assembly of the element equations into the set of global or system equations.

The finite element equations can be derived by employing either the calculus of variations or a weighted residual method to insure that the error introduced in the approximations is a minimum over the entire domain. The Galerkin weighted residual method has been successfully applied to solve a wide variety of transport problems. The basic premise of the method is that when an approximating function for the unknown values of A and Q is substituted into the governing equations, the integrated weighted residual over the domain is compelled to go to zero. The integration is performed over each element and summed to give the contributions for the whole domain for each time step (Bickford, 1990).

The finite element method has been successfully applied to problems in solid mechanics and fluid mechanics. Cooley and Moin (1976) were perhaps the first to propose a solution methodology to solve the full Saint-Venant equations using the finite element method. They found that the results obtained with the finite element method compared favorably with those obtained by others using the characteristic and the direct finite difference methods. Keuning (1976) also applied the finite element technique for solving the Saint-Venant equations. More recently, Ross et al. (1979) developed an explicit, finite element scheme for overland flow while Blandford et al. (1983) developed an implicit finite element kinematic wave model for overland flow. Burke and Gray (1983) have presented a finite element model for solving both channel and overland flow.

3.5.5. Finite Element Formulation

The finite element spatial discretization of the continuity equation is obtained using the Galerkin weighted residual method:

$$\int_{\Omega^e} \{N\} \left(\frac{\partial A}{\partial t} + \frac{\partial Q(A)}{\partial x} - q(t)\right) d\Omega^e = 0 \tag{3.35}$$

in which $\{N\} = \lfloor N_1,\ldots,N_n \rfloor^T$ the column vector of shape functions; Ω^e is the element domain; $\lfloor\ \rfloor$ symbolizes row vector; $\{\ \}$ dentoes column vector; superscript T signifies matrix transpose; and the other symbols are as previously defined. Substituting the associated approximations into Eq. (3.35) and integrating over the length of each element yields:

$$\int_0^{l_e} \{N\} \lfloor N \rfloor dx \,\{A\} + \int_0^{l_e} \{N\} \frac{d\lfloor N \rfloor}{dx} dx \,\{Q(A)\} = \int_0^{l_e} \{N\}\, q(t) \tag{3.36}$$

where

$$A = \frac{dA}{dt} = \lfloor N \rfloor \{A\}$$

$$Q(A) = \lfloor N \rfloor \{Q(A)\}$$

$$q = \lfloor N \rfloor \{q\}$$

Assuming a uniform lateral inflow, Eq. (3.36) may be expressed in matrix form as:

$$l^e \left[s \right] \{A\} + \left[f \right] \{Q(A)\} = l^e \left[s \right] q(t) \tag{3.37}$$

where:

$$\left[s \right] = \int_0^{l_e} \{N\} \lfloor N \rfloor \, dx$$

$$\left[f \right] = \int_0^{l_e} \{N\} \frac{d \lfloor N \rfloor}{dx} \, dx$$

Both element matrix evaluations can be performed analytically since they do not contain any element dependent values. Consequently, both $\left[s \right]$ and $\left[f \right]$ have been analytically integrated and included in HYDROMIN as constant matricies.

In addition to requiring a spatial approximation, Eq. (3.22) also requires a temporal discretization. In HYDROMIN the discretization of the time varying response is generated using a single step scheme. The scheme is based on knowing the flow area at time t and generating the flow area at time $t + \Delta t$ using a linear time interpolation in which Δt is the time increment. Evaluating the flow area at time $t + \Delta t$ is based on satisfying the flow equilibrium equations at time $t + \Theta \Delta t$, in which $0 < \Theta < 1$ is chosen to obtain stability and accuracy in the solution. Therefore, evaluating Eq. (3.37) at time $t + \Theta \Delta t$ gives:

$$l^e \left[s \right] \{A\}_{t+\Theta\Delta t} + \left[f \right] \{Q(A)\}_{t+\Theta\Delta t} = l^e \left[s \right] q_{t+\Theta\Delta t} \tag{3.38}$$

in which subscript $t+\Theta\Delta t$ signifies the time level at which the vector expression is evaluated. Assuming a linear variation over Δt Eq. (3.38) can be written as:

$$l^e \left[s\right]\{A\}_{t+\Theta\Delta t} = \Theta\Delta t\ l^e\ \{s\}\ q_{t+\Theta\Delta t} + l^e\left[s\right]\{A\}_t - \Theta\Delta t\ \left[f\right]\{Q(A)\}_{t+\Theta\Delta t} \qquad (3.39)$$

Eq. (3.39) is defined as the element level recurrence equation. Due to the occurrence of $\{Q(A)\}_{t+\Theta\Delta t}$ on the right hand side of Eq. (3.39), a nonzero value of Θ results in a nonlinear system of equations. Hughes (1977) has shown that for $1/2 < \Theta < 1$, the time integration scheme is unconditionally stable for nonlinear problems.

Assembly of the element equations given by Eq. (3.39) is based on continuity of the nodal flow depth and flowrate. The assembled form of Eq. (3.39) is:

$$\left[S\right]\{A\}_{t+\Theta\Delta t} = \{C\} - \Theta\Delta t\ \left[F\right]\{Q(A)\}_{t+\Theta\Delta t} \qquad (3.40)$$

where:

$$\left[S\right] = \sum_e \left[s\right]$$

$$\left[F\right] = \sum_e \left[f\right]$$

$$\{C\} = \Theta \sum_e \Delta t\ l^e\ \left[s\right]\{q\}_{t+\Theta\Delta t} + \left[S\right]\{A\}_t$$

in which $\sum_e$ represents summation over the elements; and {C} is a constant for a particular time step.

3.5.6. Kinematic Solution Methodology

For $\Theta > 0$, Eq. (3.40) is nonlinear since the right hand depends on the flow area A. Consequently, a nonlinear solution methodology is required. The kinematic wave solution may be obtained by solving Eq. (3.40) using a successive substitution approach. Using this approach, Eq. (3.40) may be written in recursive form as:

$$\left[S\right]\{A\}^{i+1}_{t+\theta\Delta t} = \{C\} - \theta\Delta t\,\left[F\right]\{Q(A^{i}_{t+\theta\Delta t})\} \tag{3.41}$$

in which the superscripts i and i+1 represent succcessive estimates of $\{A\}_{t+\theta\Delta t}$.

Eq. (3.41) is solved successively until the following convergence criterion is satisfied:

$$\left\|A^{i+1}_{t+\theta\Delta t} - A^{i}_{t+\theta\Delta t}\right\|^2_2 \,/\, \left\|A_t\right\|^2_2 < \epsilon \tag{3.42}$$

in which $\| \ \|^2_2$ is the square of the Eucledian two-norm; and ϵ is the convergence tolerance. Once Eq. (3.42) is satisfied, the value of $A_{t+\Delta t}$ can be obtained using the following equation.

$$A_{t+\Delta t} = \frac{1}{\theta}\left[A_{t+\theta\Delta t} - (1-\theta)A_t\right] \tag{3.43}$$

To initiate the solution of Eq. (3.41), an initial estimate of the nodal flow area must be obtained for time $t+\theta\Delta t$. In HYDROMIN this estimate is obtained using an explicit finite difference approximation of Eq. (3.22). For node j, this approximation may be expressed as (Blandford and Meadows, 1990):

$$A^{j}_{t+\theta\Delta t} = A^{j}_{t} + \theta\Delta t\left[q_t - \frac{(Q^{j}_{t} - Q^{j-1}_{t})}{l_e}\right] \tag{3.44}$$

3.5.7. Diffusion Wave Solution Methodology

In order to apply the previous solution methodology to the diffusion wave form of Eq. (3.29), Eq. (3.41) must be modified to reflect the dependency of Q on both A and S (the friction slope). This may be expressed as:

$$\left[S\right]\{A\}^{i+1}_{t+\theta\Delta t} = \{C\} - \theta\Delta t\,\left[F\right]\{Q(A^{i}_{t+\theta\Delta t}, S^{i}_{t+\theta\Delta t})\} \tag{3.45}$$

This equation may be solved in the same manner as Eq. (3.41) with the friction slope also expressed as a function of the nodal flow areas.

During each iteration of Eq. (3.45) the friction slope over each finite element is updated as a function of the new values of the nodal flow areas. The friction slope associated with each node is obtained using a backward finite difference scheme. For the case with a rectangular rill with a uniform width the friction slope associated with node j at time $t+\Theta\Delta t$ may be obtained as follows:

$$S^{i}_{t+\theta\Delta t} = S_o - \frac{A^{j}_{t+\theta\Delta t} - A^{j-1}_{t+\theta\Delta t}}{w_r l^e} \tag{3.46}$$

where S_o = the subrill bottom slope, w_r = the subrill width, and $A^{j}_{t+\theta\Delta t}$ and $A^{j-1}_{t+\theta\Delta t}$ are nodal flow areas associated with iteration i.

3.5.8. Rill Network Solution Methodology

The solution of the diffusion wave equations for a network of subrills results in an additional depth continuity constraint for each junction node in the network. These constraints are satisfied for each time step using a dissaggregated dual level iteration scheme. The lower level of the iteration procedure involves the iterative solution of the nodal flow areas. At this level, each individual subrill is evaluated one at a time until the associated set of nodal flow areas have been determined.

The upper level of the iteration strategy involves the enforcement of the flow depth continuity at each junction node. This is accomplished by setting the downstream depth boundary condition for all subrills before they are passed back down to the lower iteration level.

During each upper level iteration, the subrills that drain into each junction node are identified. The downstream flow depths associated with these subrills are updated by equating them to the upstream flow depth associated with the subrill that drains the associated junction. Mathematically, this relationship may be expressed as:

$$h^{in}_{t+\theta\Delta t} = h^{j1}_{t+\theta\Delta t} \quad \text{for } all \ i \in \{j\} \ \text{for } all \ j \tag{3.47}$$

where $h^{in}_{t+\theta\Delta t}$ = the depth at time $t+\Theta\Delta t$ associated with the last (downstream) node in each upstream subrill i (where n = the number of finite element nodes in each subrill) and $h^{j1}_{t+\theta\Delta t}$ = the depth at time $t+\Theta\Delta t$ associated with the first (upstream) node in the downstream subrill j, where j also represents the index of the junction drained by the subrill and $\{j\}$ = the set of subrills that drain into junction j. For a network with rectangular cross sections, this relationship may be expressed as:

$$A^{in}_{t+\theta\Delta t} = \frac{A^{j1}_{t+\theta\Delta t}}{w_j} w_i \quad \text{for } all \ i \in \{j\} \ \text{for } all \ j \tag{3.48}$$

where A = the nodal flow area, and w = the subrill bottom width.

For a particular time step, the upper level iteration procedure converges when the following condition is satisfied for each junction node in the network:

$$\frac{\frac{1}{N_j}\sum_{i=1}^{N_j} h^{in}_{t+\theta\Delta t} - h^{j1}_{t+\theta\Delta t}}{h^{j1}_{t+\theta\Delta t}} < \epsilon \tag{3.49}$$

where N_j = the number of subrills that drain into junction j and ϵ = a user specified convergence level. Again, for a network with a rectangular channel this condition may be expressed as:

$$\frac{\frac{1}{N_j}\sum_{i=1}^{N_j} A^{in}_{t+\theta\Delta t}/w_i - A^{j1}_{t+\theta\Delta t}/w_j}{A^{j1}_{t+\theta\Delta t}/w_j} < \epsilon \tag{3.50}$$

3.5.9. Boundary Conditions

Before the previously discussed dual level iteration procedure can be initiated, the intial values of all nodal areas at time $t = 0$ must be specified. In the proposed algorithm, these values are obtained by performing a steady state profile analysis for the entire network system. The profile analysis uses the same solution methodology as employed by the HEC2 computer program. This methodology is based on solving the following two equations at each finite element node using an iterative procedure:

$$WS_2 + \frac{V_2^2}{2g} = WS_1 + \frac{V_1^2}{2g} + h_e \tag{3.51}$$

$$h_e = L\overline{S} + \frac{V_2^2}{2g} - \frac{V_1^2}{2g} \tag{3.52}$$

where:

$WS_1,\ WS_2$ = water surface elevations at ends of reach

$V_1,\ V_2$ = mean velocities (total discharge / total flow area) at ends of reach

g = acceleration of gravity

h_e = energy head loss

L = reach length

$\overline{S}$ = the average representative friction slope for the reach

In addition to the initial boundary conditions at time $t = 0$, upstream, downstream, and lateral inflow boundary conditions must be specified for the entire network for all time steps. The downstream boundary condition for the network (i.e., the downstream flow area for the last node of the lowest subrill) is set based on an assumption of normal flow depth at the network outlet. The upstream boundary condition for each of the first order subrills is established from the associated source area hydrographs. The lateral inflow boundary condition for each element of each subrill is set using the associated interrill element hydrographs.

3.5.10 Algorithm Summary

The specific algorithms for both the upper level and lower level iteration schemes are outlined in the following sections.

3.5.10.1 Upper Level Algorithm

The upper level of the proposed dual level disaggregated iteration strategy may be summarized as follows:

(0) Perform a steady state profile analysis in order to determine $\{A\}_t$ and $\{Q\}_t$ for t = 0 for each finite element node associated with each subrill in the network.

(1) Set the upstream, downstream, and lateral inflow boundary conditions for the next time step.

(2) Perform the lower level iteration loop in order to determine the set of nodal parameter values (i.e., $A_{t+\Theta\Delta t}$, $S_{t+\Theta\Delta t}$, and $Q_{t+\Theta\Delta t}$) for each subrill for time $t+\Theta\Delta t$.

(3) Evaluate the depth continuity requirement for each junction node in the network using Eq. (3.49).

(4) If Eq. (3.49) is satisfied for all junction nodes, continue, otherwise update the downstream boundary conditions for all subrills (exclusive of the last subrill which is set in step (1)) using Eq. (3.47) and return to (2).

(5) Determine the set of nodal parameter values (i.e., $A_{t+\Delta t}$, $S_{t+\Delta t}$, and $Q_{t+\Delta t}$) for each subrill for time $t+\Delta t$ using Eqs. (3.43) and (3.28).

(6) Stop once all time steps have been evaluated, otherwise update the nodal parameter values and return to (1).

3.5.10.2 Lower Level Algorithm

The lower level of the proposed dual level disaggregated iteration strategy may be summarized as follows:

(1) Estimate the intial values of $A_{t+\theta\Delta t}$ using Eq. (3.44).

(2) Determine the value of $S_{t+\theta\Delta t}$ using Eq. (3.46).

(3) Determine the value of $Q_{t+\theta\Delta t}$ using Eq. (3.28).

(3) Solve for $A_{t+\theta\Delta t}$ using Eq. (3.45).

(4) Check for convergence using Eq. (3.42).

(5) If convergence is achieved then stop, otherwise update the flow area iteration value (ie., $A^{i}_{t+\theta\Delta t} = A^{i+1}_{t+\theta\Delta t}$) and go to (2).

IV. MODEL VALIDATION

This chapter contains the results of a series of validation studies for the hydrologic model HYMODRIN. In order to test the applicability of HYMODRIN in modeling the hydrologic response of dynamic rill networks, the model was evaluated using three separate model validation studies. Each study is discussed in detail in the following sections.

4.1. HYDROLOGIC VALIDATION

The first validation study involved the evaluation of the hydrologic components of the model. These included both the Green Ampt infiltration routine and the nonlinear reservoir overland flow routine. In order to test the hydrologic components of HYMODRIN, the model was applied to two small watersheds. Each watershed was simulated for two different rainfall events. Data for the two watersheds, identified as Hastings 4H and 5H were obtained from a previous hydrologic study by Blandford et al. (1983).

The two watersheds are shown in Figures 4.1 and 4.2. Both watersheds are relatively flat at the upstream end, but converge somewhat toward a definite flow path. Each watershed was modeled as a single plane with constant slope and constant width. The calculated values are shown on the associated figure.

For the rainfall events selected, both watersheds were classified as pasture land and both consisted of the same three soil types: Hastings silt loam, Hastings silty clay loam, and Colby silt loam. Each soil occurred in layers, but infiltration did not proceed beyond the first layer (5 inches) in any of the storm events analyzed. Therefore, properties are given only for the first layer of each soil type.

The Green Ampt parameters for each soil were obtained from Meadows et al. (1983). They determined the wetting front suction , ψ_f, after Brakensiek (1977), with the Brooks and Corey (1966) parameters determined from desorption data reported by the ARS. The ARS data included the moisture content, at various soil depths, corresponding to capillary pressures of 0.1, 0.3, 0.6, 3.0 and 15.0 bars. The porosity ϕ, of each soil was determined from ARS bulk density data. Table 4.1 list values of ϕ and ψ_f for each soil type. Table 4.1 also shows the composition of both watersheds according to percent of each soil type.

The nonlinear reservoir Green and Ampt analyses were performed for four different rainfall events; two on each watershed. The rainfall data for each of the events is summarized in Table 4.2. Table 4.3 presents the calculated hydraulic conductivities, K_s, for each storm.

For each rainfall event, the model was calibrated by adjusting the Manning's roughness coefficient n along with the intial moisture content. In each case the model parameters were adjusted so as to match the observed and predicted peack discharges while at the same time maintaining continuity of volume between the observed and predicted hydrographs. The calibrated parameter values are also shown in Table 4.3.

The results of the validation analysis are shown in Figures 4.3 - 4.6. Although the volumes of the computed and observed hydrographs were found to match very closely, two of the computed peak discharges were less than the observed corresponding values. The timing correlations between the observed and predicted hydrographs were also variable. It is possible that a significant part of this variability may be attributed to the spatial or temporal variability of the rainfall data. For both watersheds, the associated rain gage was located outside the watershed boundaries.

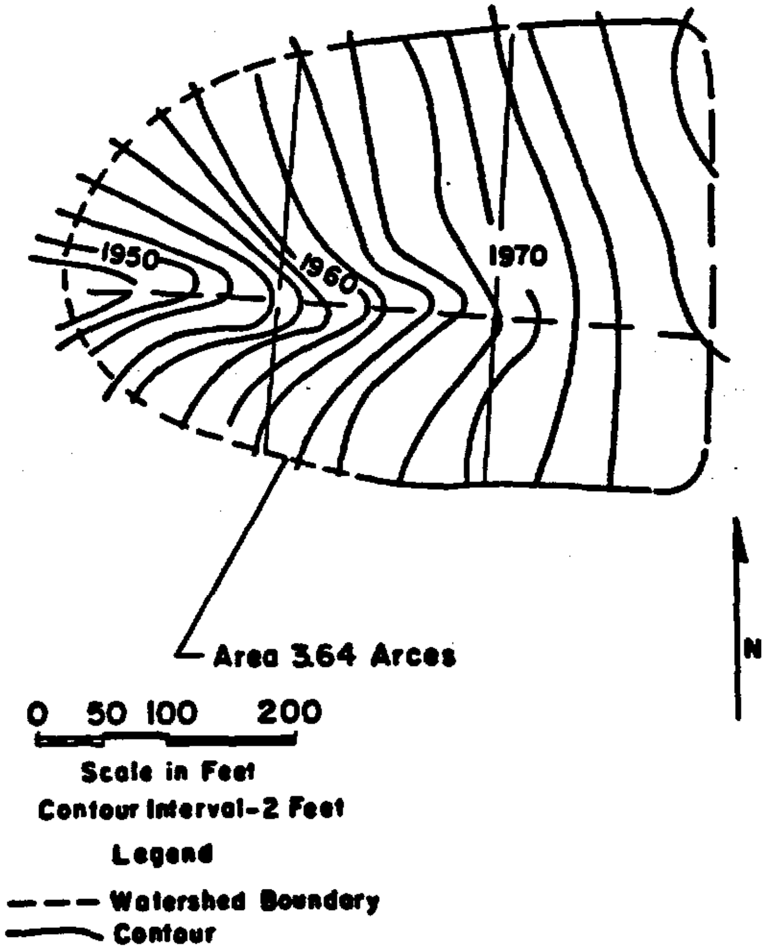

Figure 4.1 Hastings 4H Watershed

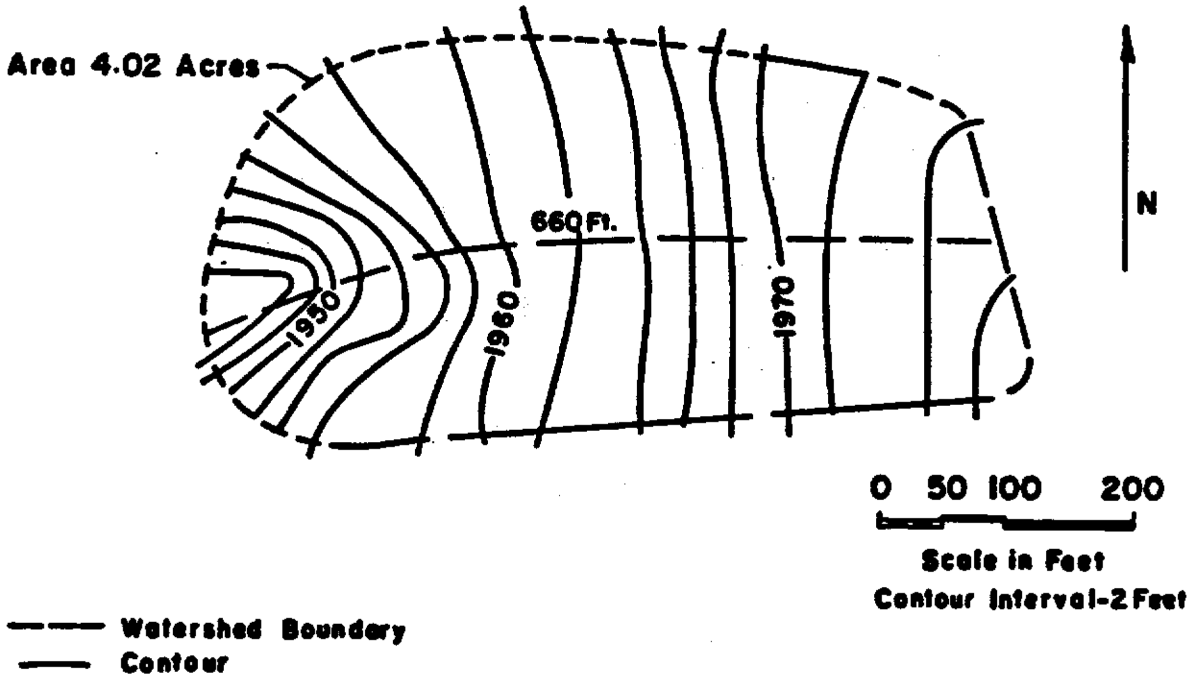

Figure 4.2 Hasings 5H Watershed

Table 4.1 Soil Characteristics of Example Watersheds

Soil Type:	Hastings Silt Loam	Hastings Silty Clay Loam	Colby Silt Loam
ϕ	.5509	.630	.550
ψ_f (ft)	.812	.787	1.780
% of 4H	65	26	9
% of 5H	87	7	6

Table 4.2 Summary of Rainfall Data for Hastings Events

Watershed 5H August 11, 1961		Watershed 5H July 26, 1964	
Time (hr. min)	Accumulated Rain (in)	Time (hr. min)	Accumulated Rain (in)
0.26	0.00	16.44	0.00
0.38	0.74	16.51	0.78
0.42	0.76	16.55	1.07
0.52	1.50	16.58	1.28
1.00	1.61		
1.30	1.70		
1.50	1.71		

Watershed 4H May 4, 1959		Watershed 4H June 21, 1964	
Time (hr. min)	Accumulated Rain (in)	Time (hr. min)	Accumulated Rain (in)
14.18	0.00	5.03	0.00
14.20	0.03	5.06	0.31
14.22	0.10	5.10	0.60
14.24	0.22	5.14	0.90
14.26	0.40	5.19	1.00
14.28	0.52	5.36	1.07
14.30	0.60		
14.32	0.64		
15.07	0.75		
15.27	0.76		

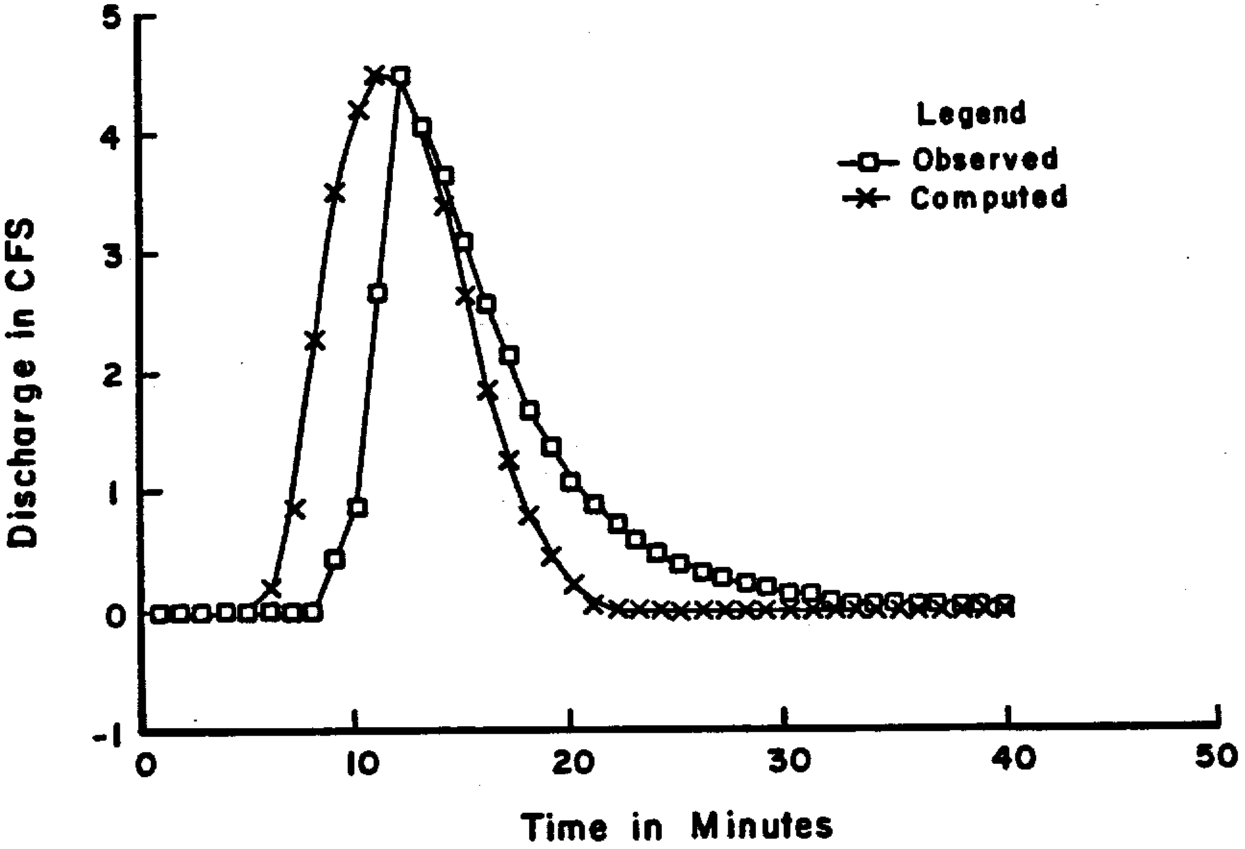

Figure 4.3 Hastings 4H Hydrographs for (5/4/59)

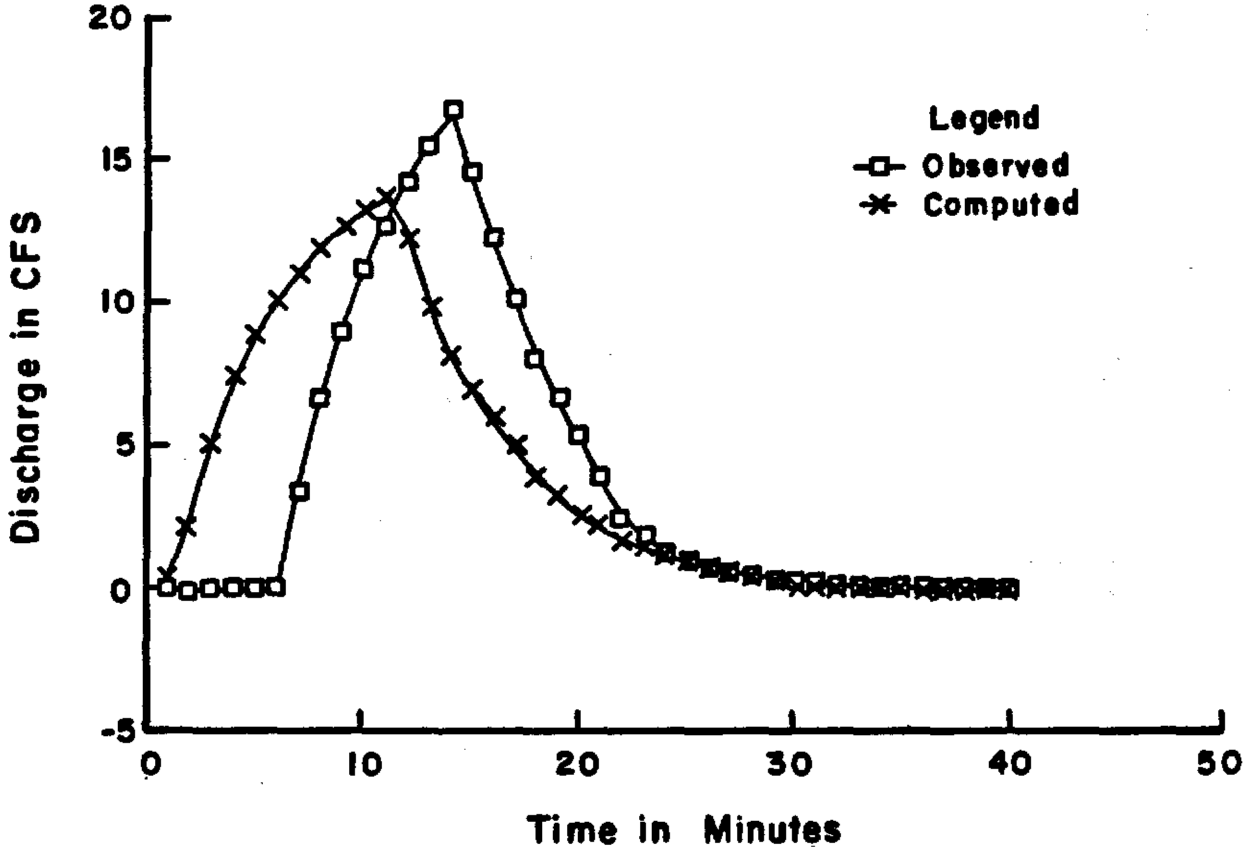

Figure 4.4 Hastings 4H Hydrographs for (6/21/64)

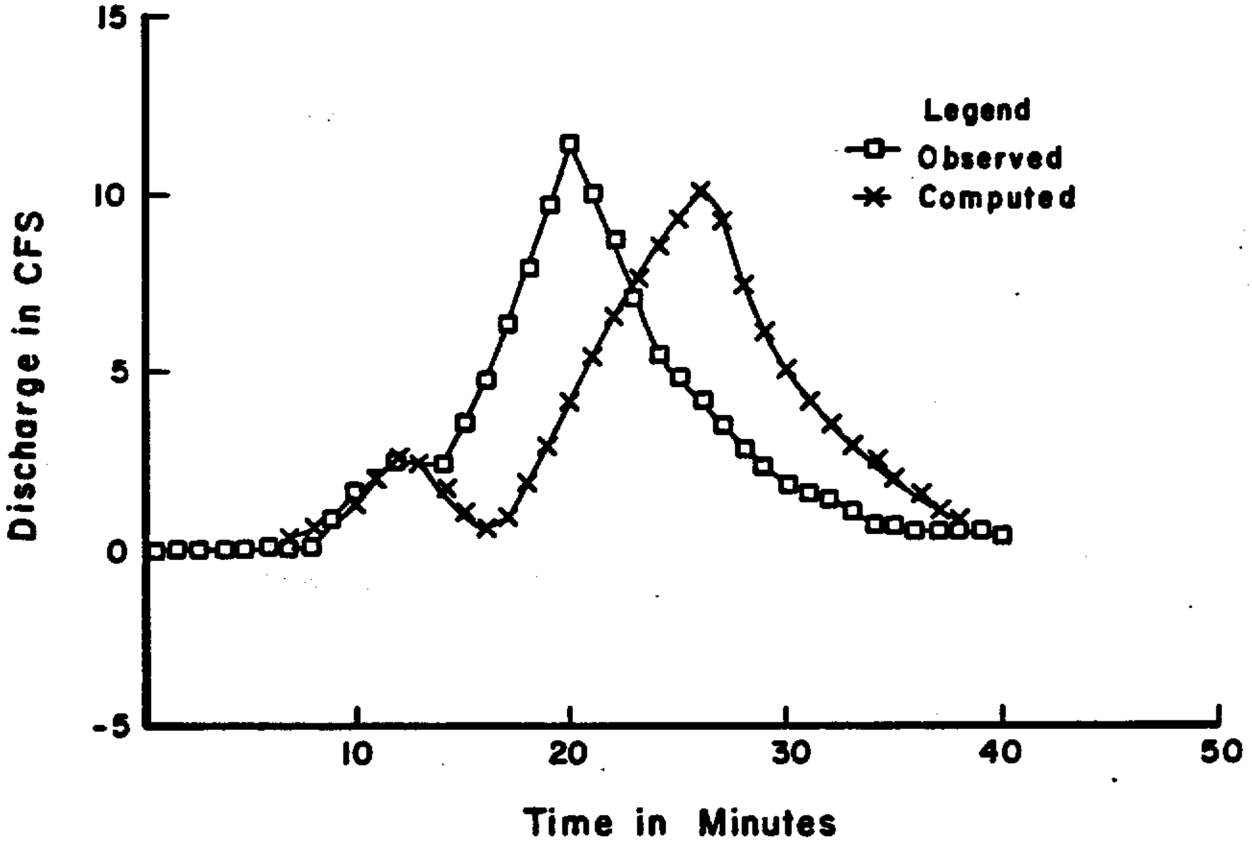

Figure 4.5 Hastings 5H Hydrographs for (8/11/61)

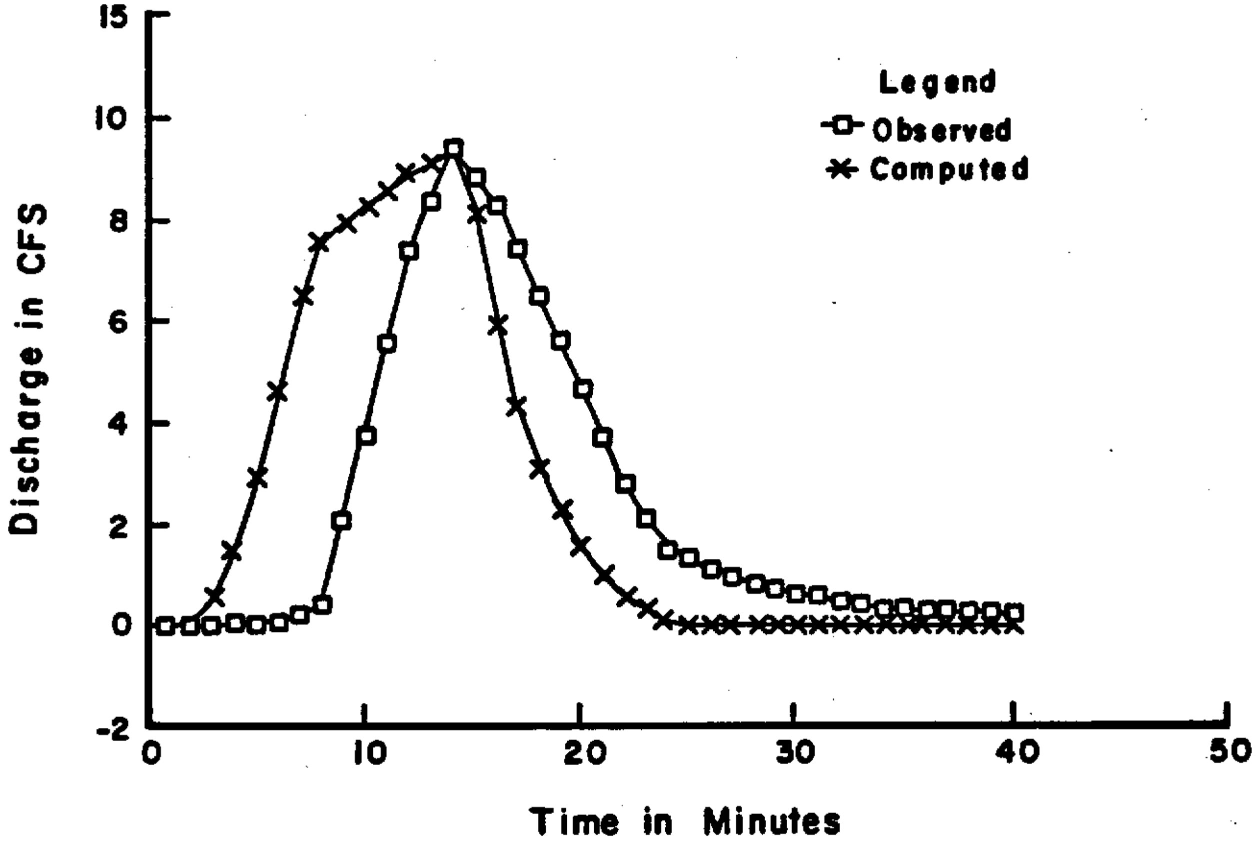

Figure 4.6 Hastings 5H Hydrographs for (7/26/64)

4.2 HYDRAULIC VALIDATION

The second validation study involved the evaluation of the finite element diffusion wave routing algorithm. In order to test the routing algorithm it was applied to a hypothetical network consisting of two junctions and six branches (see Figure 4.7). This network was previously evaluated using a dynamic wave finite difference model by Akan and Yen (1981). All branches are assumed to be rectangular in cross sections. The dimensions of the channels are listed in Table 4.4.

The hypothetical hydrographs that are routed through the channel network in the two examples are shown in Figures 4.8 and 4.9, The intial condition for both examples was a steady flow condition corresponding to a discharge of 3 m^3/s (106 cfs) in channels 1 and 4, 2 m^3/s (71 cfs) in channels 2 and 3, 7 m^3/s (247 cfs) in channel 5, and 10 m^3/s (353 cfs) in channel 6. The downstream boundary condition at the exit of channel 6 was specified using the uniform flow equation assuming the channel is hydraulically long. A time increment of $\Delta t = 60$ *sec* was used in both examples. Each channel was modeled using three linear finite elements.

The computed discharge hydrographs for both examples are shown in Figures 4.10 and 4.11. As can be observed from these figures, the results of the proposed diffusion-wave model are in good agreement with those of the dynamic model. The outflow hydrographs computed for channels 2 and 3 in Example I and shown in Figure 4.10b clearly demonstrate the effects of downstream flow conditions. In Example I, a constant upstream inflow of 2 m^3/s (71 cfs) equal to the baseflow rate is used for channels 2 and 3 as shown in Figure 4.8. As a result, a steady flow condition will prevail in both channels 2 and 3 if the downsteam flow conditions do not exist. However, in this case, the flood wave traveling through channel 1 raises the water surface at the junction where branches 1, 2 and 3 join. This decreases the hydraulic gradient in channels 2 and 3 causing discharges lower than the constant upstream inflow rate. As a result, from continuity requirements, the channel storage is increased during the period of low discharge. As the downstream backwater recedes with time, the excess water is released from the channel storage, and discharges higher than the constant inflow occurs until the steady state condition is again reached asymptotically.

Table 4.3 Hastings Model Parameters

Watershed	Event	K_s	θ_i	n
4H	May 4, 1959	.36	.50	.029
4H	June 21, 1964	.15	.55	.025
5H	August 11, 1961	.58	.15	.025
5H	July 26, 1964	.58	.46	.030

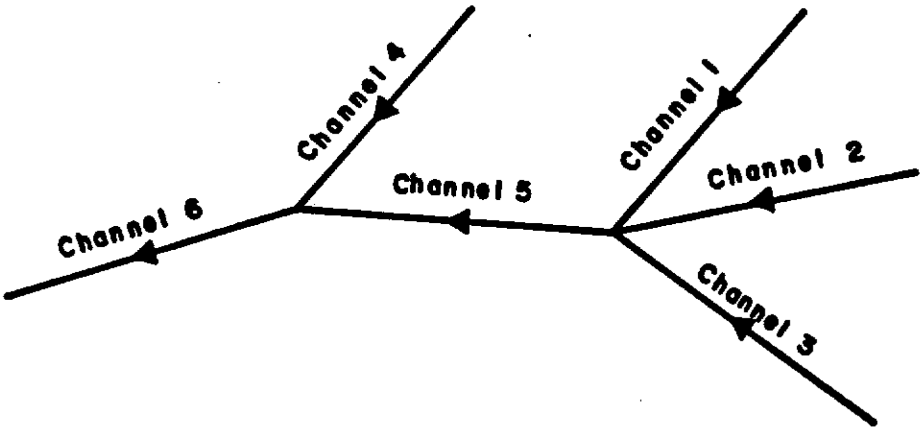

Figure 4.7 Hypothetical Channel Network

Table 4.4 Physical Properties of Hypothetical Channel Network

Channel number (1)	Length, in feet (meters) (2)	Slope (3)	Width, in feet (meters) (4)	Manning n (5)
1	1968.5 (600)	0.0005	16.4 (5)	0.0138
2	1968.5 (600)	0.0005	16.4 (5)	0.0207
3	1968.5 (600)	0.0005	16.4 (5)	0.0207
4	1968.5 (600)	0.0005	16.4 (5)	0.0138
5	1968.5 (600)	0.0010	26.2 (8)	0.0141
6	1968.5 (600)	0.0010	32.8 (10)	0.0125

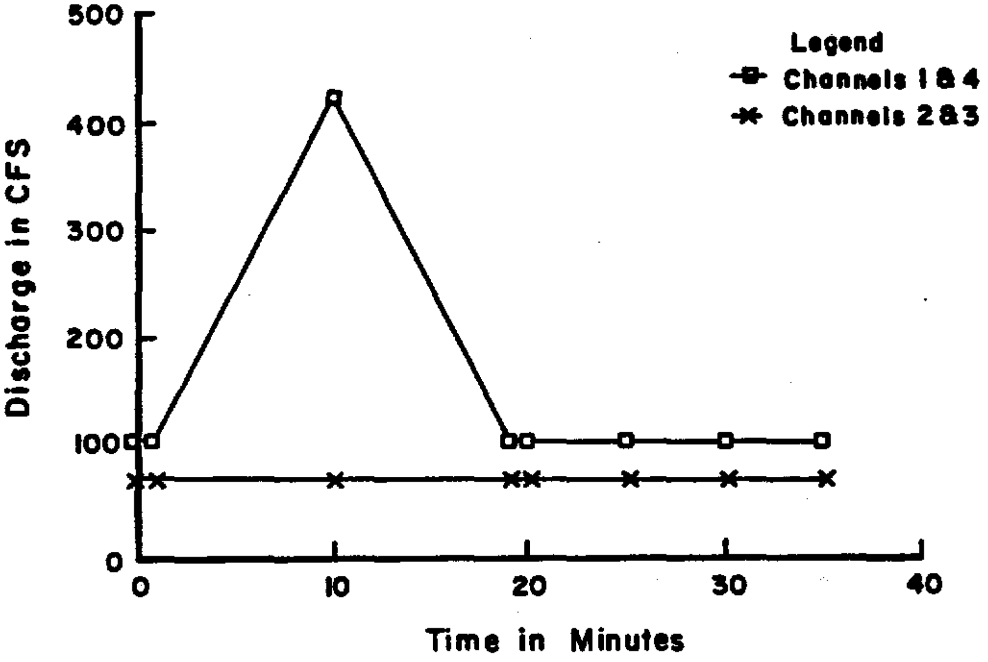

Figure 4.8 Inflow Hydrographs for Example I

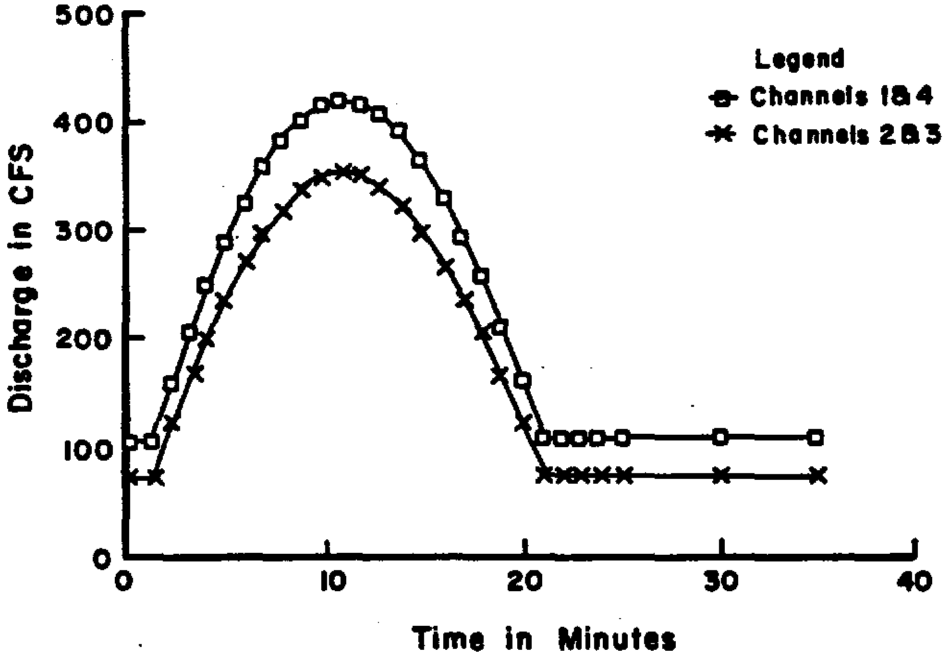

Figure 4.9 Inflow Hydrographs for Example II

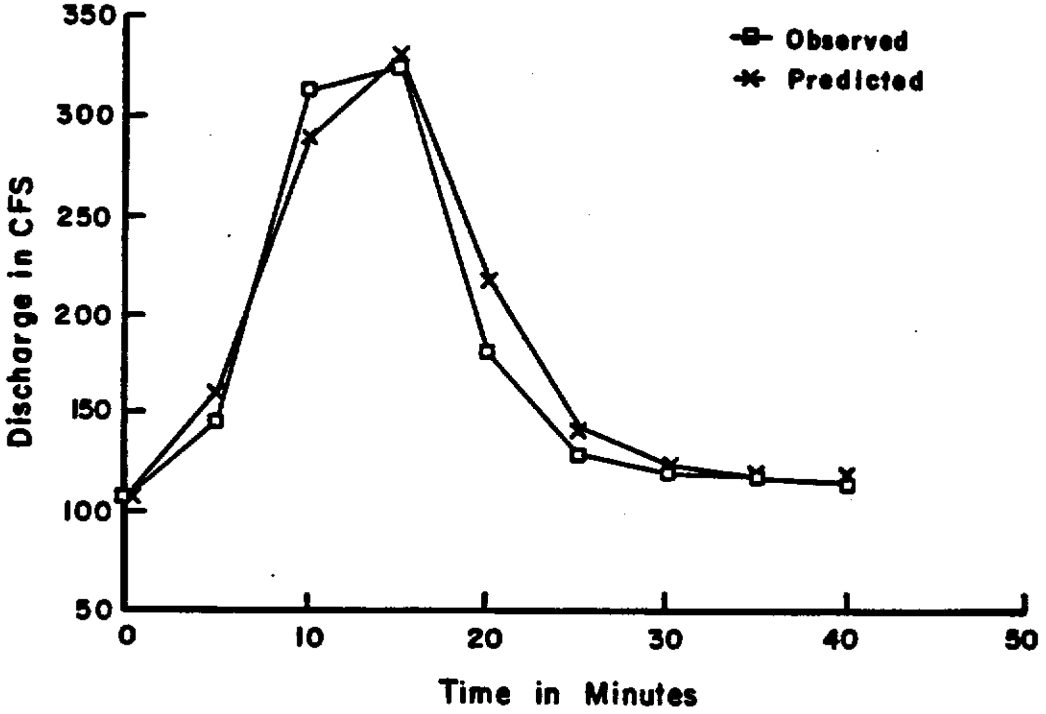

Figure 4.10a Example I Hydrographs for Channel 1

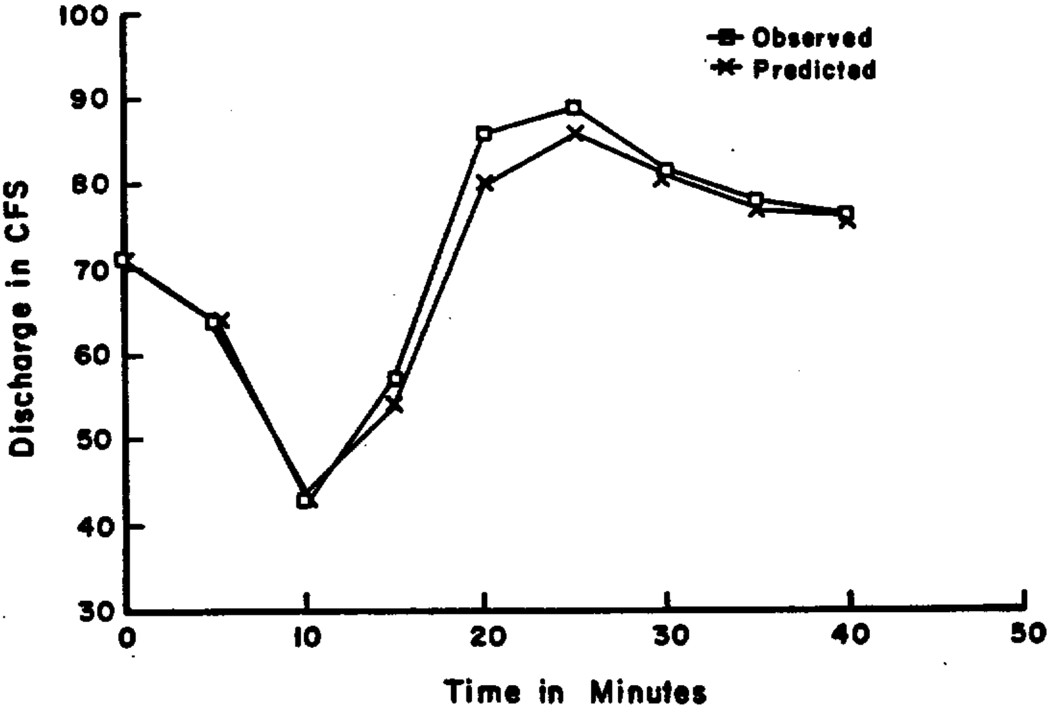

Figure 4.10b Example I Hydrographs for Channels 2 and 3

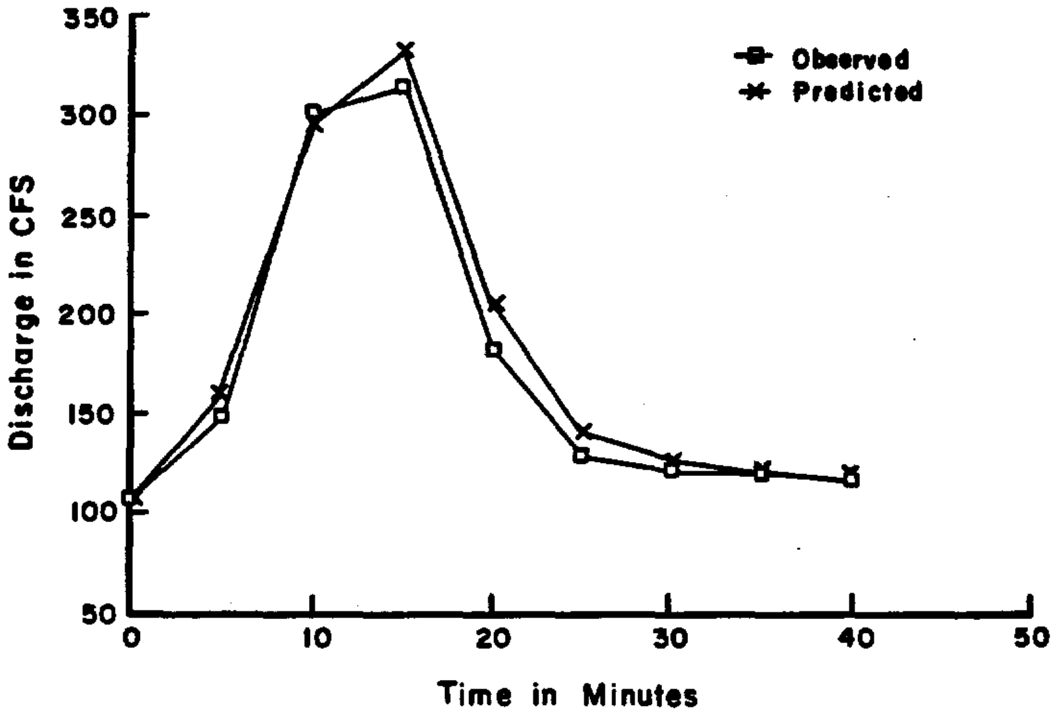

Figure 4.10c Example I Hydrographs for Channel 4

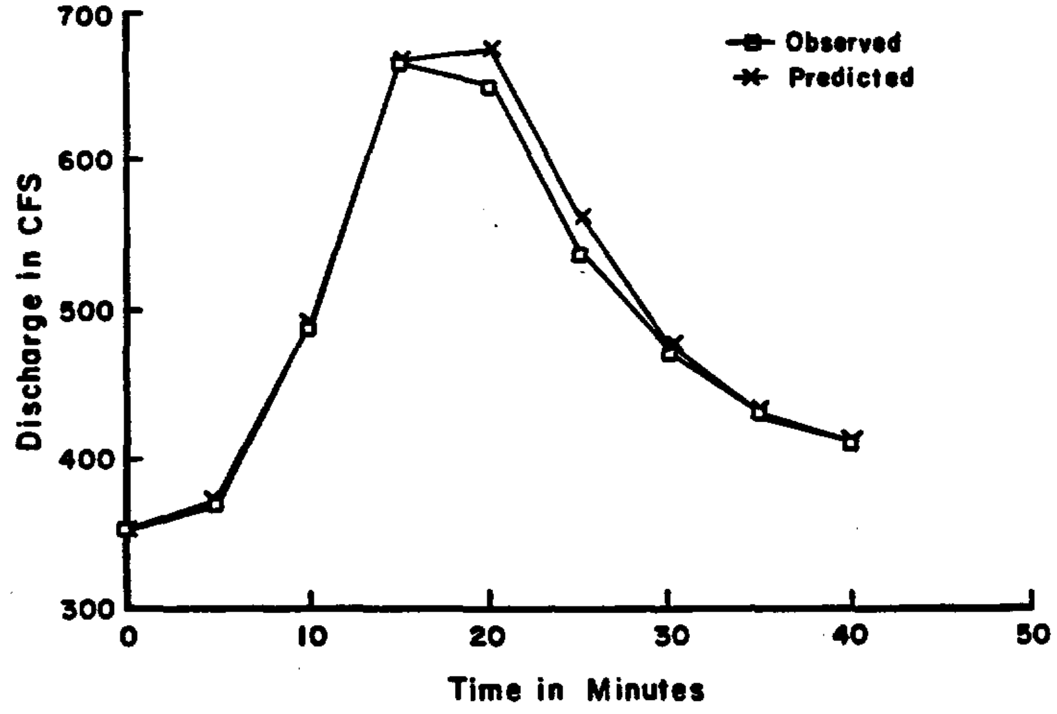

Figure 4.10d Example I Hydrographs for Channel 6

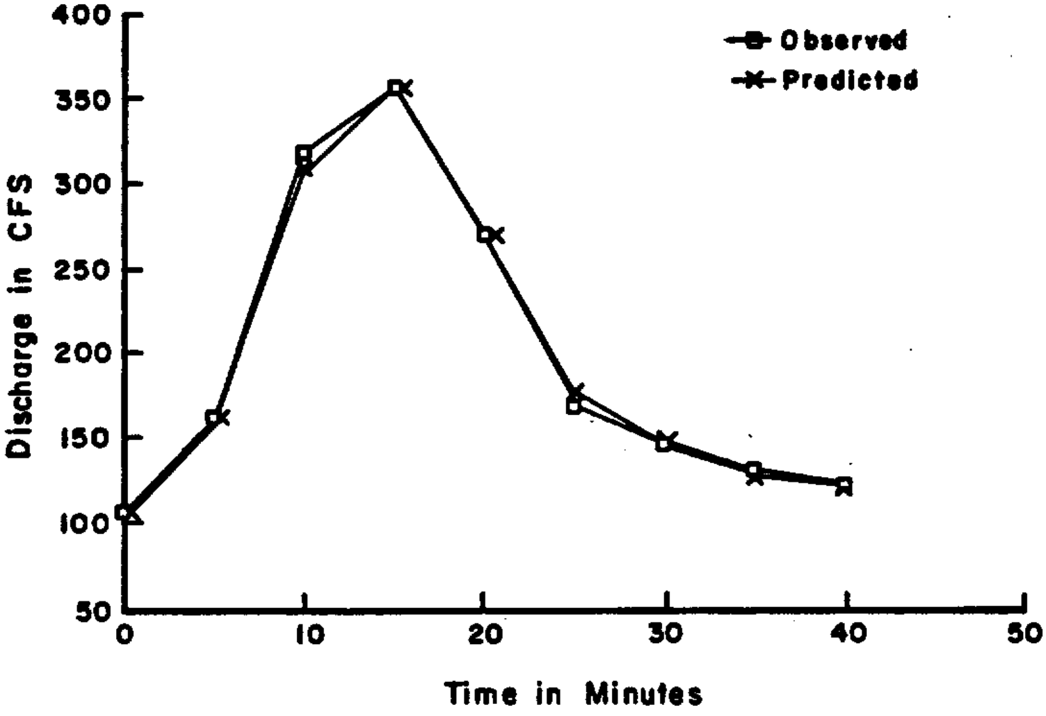

Figure 4.11a Example II Hydrographs for Channel 1

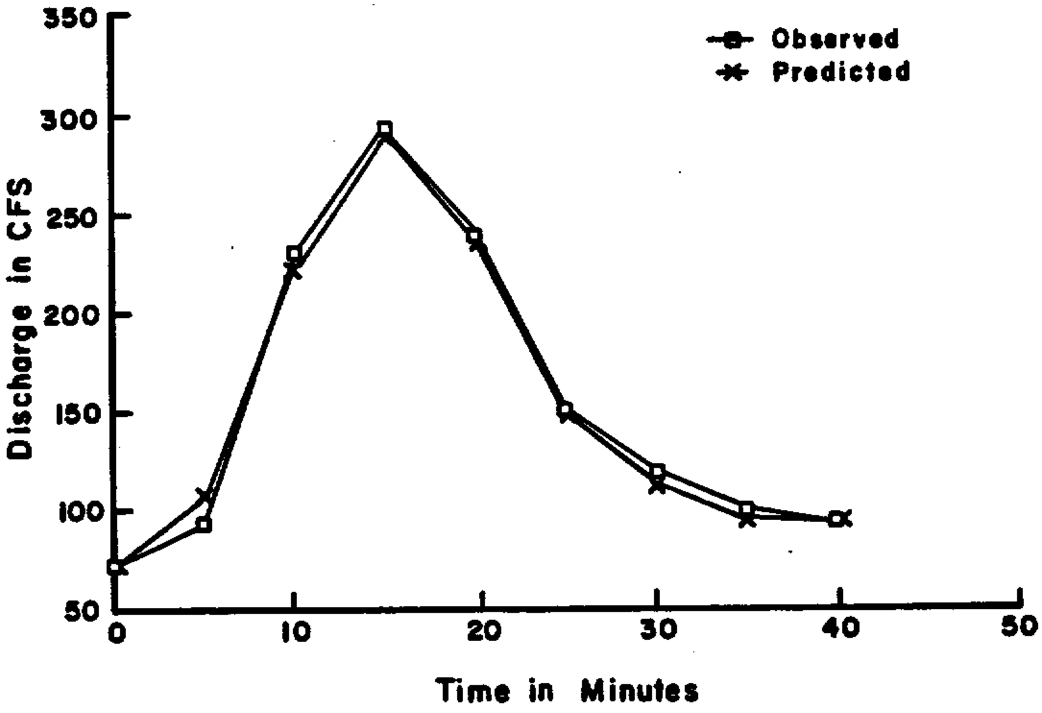

Figure 4.11b Example II Hydrographs for Channels 2 and 3

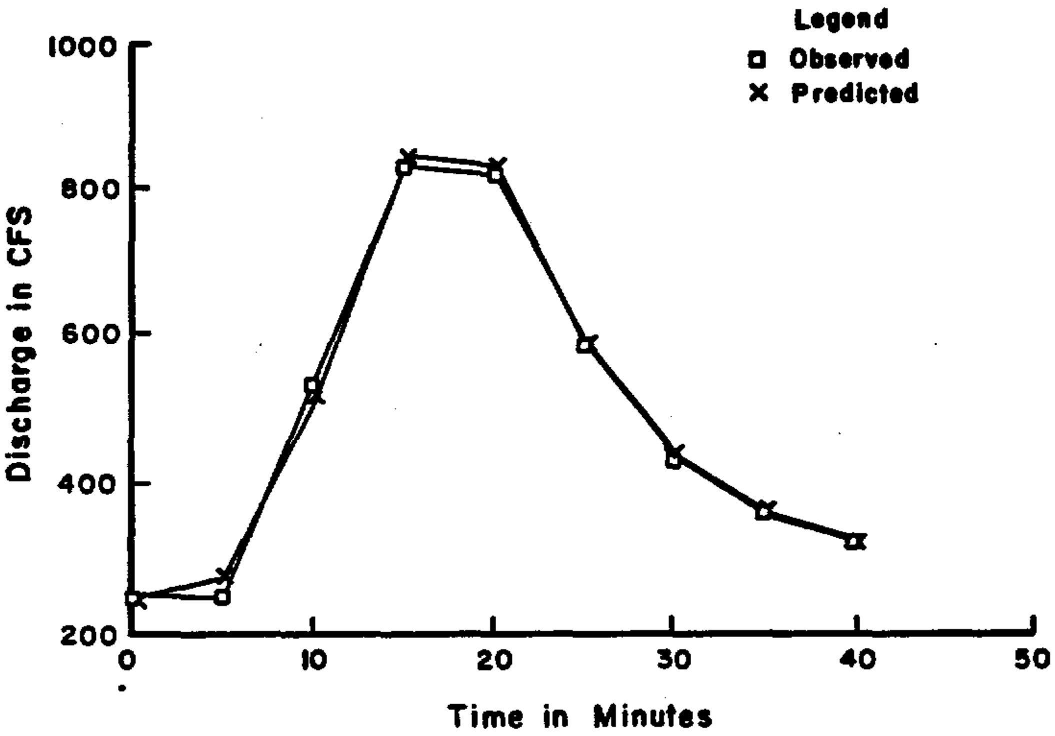

Figure 4.11c Example II Hydrographs for Channel 5

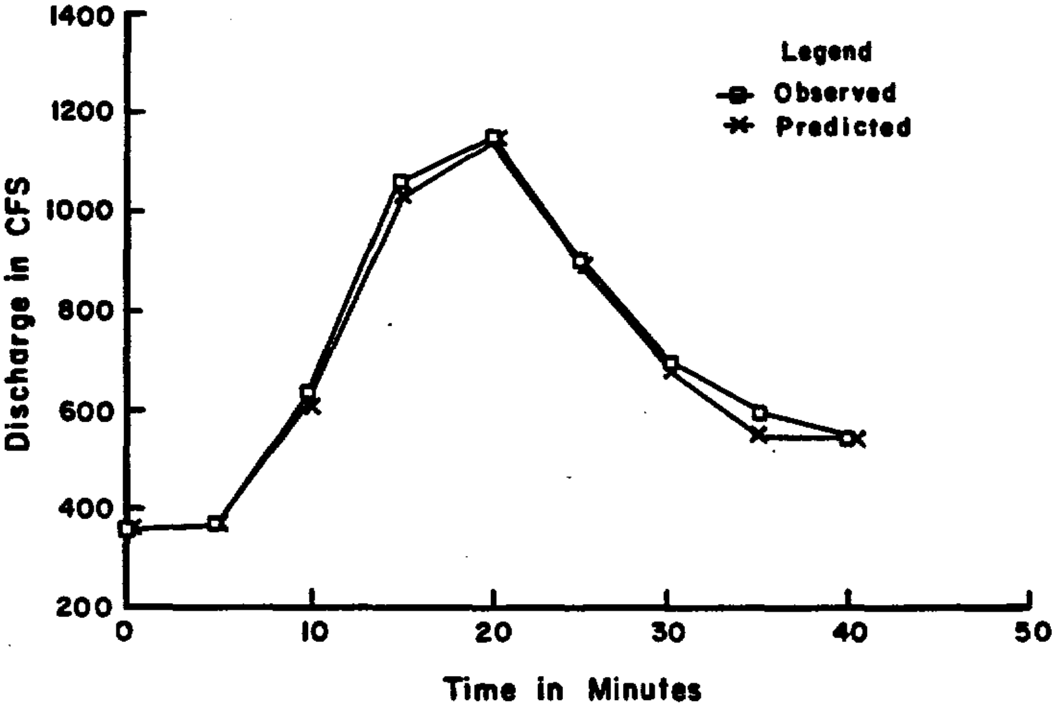

Figure 4.11d Example II Hydrographs for Channel 6

4.3. DYNAMIC RILL VALIDATION

The final validation study involved the evaluation of the complete hydrologic model. In order to test the complete model structure, HYMODRIN was applied to an example rill network system. The rill network system is shown in Figure 4.12. The CDATA and HDATA files for the entire system are provided in APPENDIX E. Each subrill was modeled as a rectangular channel.

The physical characteristics of the soils in the rill system are shown in Table 4.5. Values of the capillary suction pressure ψ_f and the hydraulic conductivity K_s (cm/sec) were obtained ined using the following relatioships developed by Brakensiek and Engleman (1979):

$$\ln \psi_f = 3.4948 - 0.0146\ (\%sand): \quad r^2 = -0.874 \tag{4.1}$$

$$\ln K_s = -11.9661 - 1.9784 \ln (\%clay/100): \quad r^2 = -0.982 \tag{4.2}$$

Average values of the bulk density (g/cm^3), depth to the impermeable layer (in), and the intial soil moisture (%) were obtained by averaging 72 sample values from each of the experimental plots. The bulk density and soil moisture values were obtained using a bulk density meter while the depth to the impermeable layer values were obtained using a pentrometer. An estimate of the soil porosity (ϕ) was obtained in terms of the bulk density (B_d) using the following equation:

$$\phi = 1 - \frac{B_d\ (g/cm^3)}{2.65\ (g/cm^3)} \tag{4.3}$$

In applying HYMODRIN to the example rill system, each subrill was simulated using a series of linear elements. The number of elements was dependent upon the length of the particular subrill. The maximum number of elements used to simulate any one subrill was limited to ten.

HYMODRIN was applied to the example rill system using a computational time interval of 10 seconds. The rill system was analyzed for a single rainfall event which consisted of a constant rainfall intensity of 87 mm/hr and a storm duration of 90 minutes.

In applying HYMODRIN to the example rill system a numerical instability problem was encountered in processing some of the subrill elements. Attempts to isolate the cause of the problem proved to be unsuccessful. As a result, the entire network system could not be simulated.

In order to evaluate the mass continuity components of the model, the rill network was simulated as a single plane element using the watershed characteristics indentified in Table 4.5. The model was then calibrated by adjusting the Manning's n coefficient, the initial moisture content, and the hydraulic conductivity. The resulting hydrograph is shown in Figure 4.13. As can be seen from the figure, both the observed and predicted hydrographs are very similar.

Table 4.5 Watershed Characteristics of Example Rill Network

Depression Storage (mm)	10.0
Initial Soil Moisture (%)	10
Porosity (%)	52
Capillary Suction Pressure (cm)	25
Hydraulic Conductivity (cm/hr)	.700
Soil Depth (cm)	15.5
Slope (%)	9
Mannings Roughness	.02

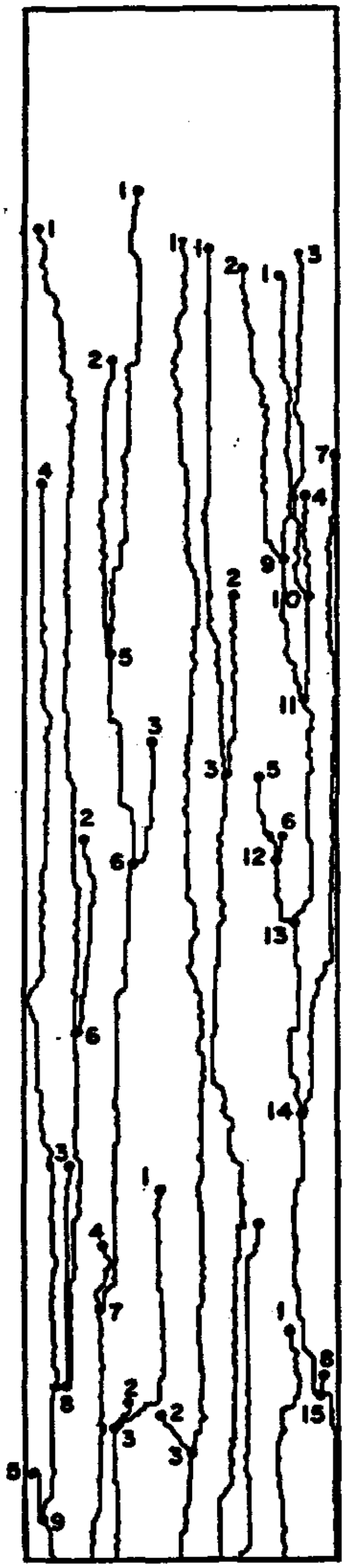

Figure 4.12 Schematic of Example Rill Network

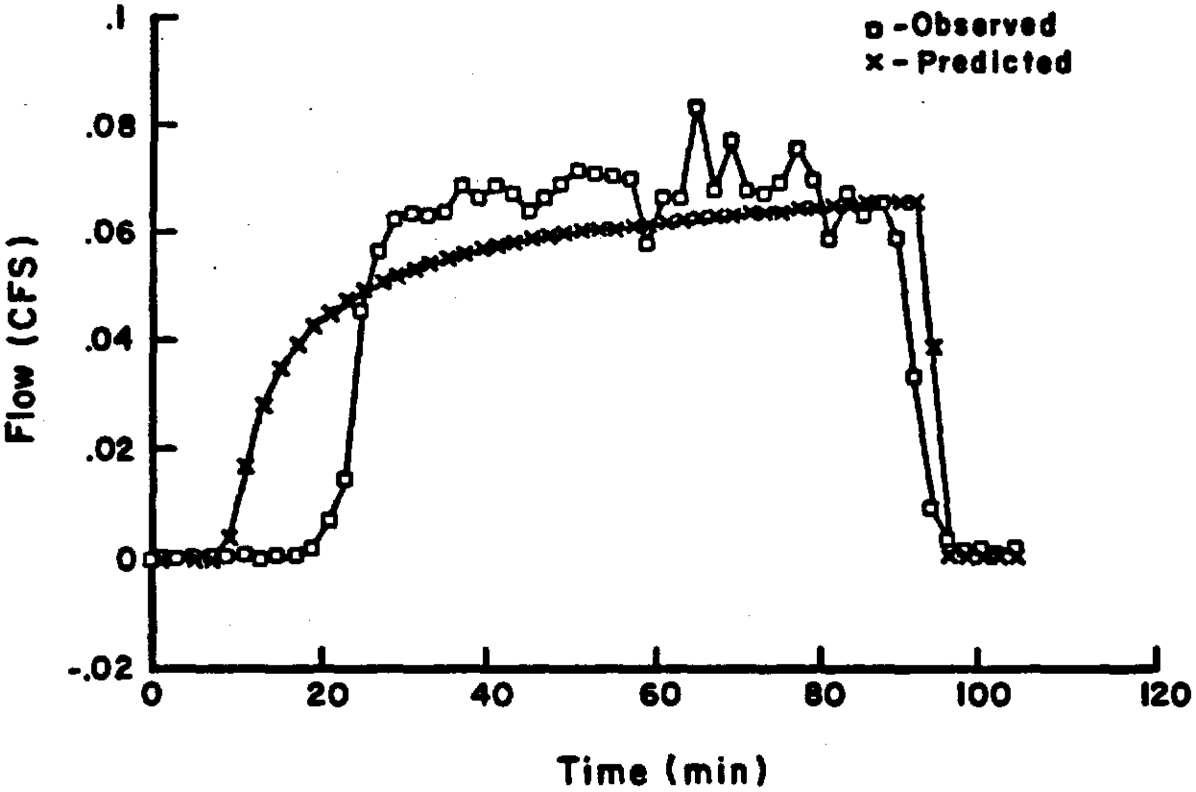

Figure 4.13 Hydrograph for Example Rill Network

V. SUMMARY AND RECOMMENDATIONS

A comprehesive model has been developed for use in modeling the hydrologic response of rill network systems. The model is composed of both a hydrologic runoff component and a hydraulic channel routing component. The hydrologic component of the model uses a Green Ampt infiltration approach linked with a nonlinear reservoir runoff model. The channel routing component of the model is based on a finite element solution of the diffusion wave equations. In order to account for backwater effects the model employs a dual level iteration scheme.

The individual components of the model for evaluated for two separate watersheds and a hypothetical channel network. In each case the model was able to reproduce either the observed or documented results with a satisfactory level of performance.

After the individual components of the model were tested the entire model was applied to an example rill network system. Unfortunately, a numerical instability problem was encountered which prevented the processing of the entire rill network system. As a result, the network was modeled as a single overland plane. The hydrograph which resulted from the application of the model in this model was in close correlation with the observed hydrograph.

The existing model needs further testing and evaluation before application in a production setting. The main area of concern lies in the sensitivity of the diffusion wave solution methodology to computer round-off errors. Future work with the model should focus on the underlying reason for the instability of the equations with a goal of producing a more stable algorithm.

REFERENCES

Akan, O. A., and Yen, B. C., Diffusion Water Flood Routing in Channel Networks, *Journal of the Hydraulics Division,* ASCE, Vol. 107, No. HY6, pp. 719-730.

Al-Mashidani, G., and Taylor, C., 1974, Finite Element Solutions of the Shallow Water Equations: Surface Runoff, Finite Elements in Flow Problems, UAH Press, Huntsville, AL.

Bickford, W.B., 1990, A First Course in the Finite Element Method, Irwin Press, Inc., Boston, MA.

Blandford, G.E., Peters, N. II, and Meadows, M.E., 1983, Development of Models for Simulating Stormwater Runoff from Surface Coal Mined Lands, Dept. of Civ. Eng., Univ. of Kentucky, Lexington, Kentucky, Final Report, *Rep. No. G5115213,* Prepared for U.S. Dept. of Interior, Office of Surface Mining, Div. of Res., Washington D.C., pp. 130.

Blandford, G.E., and Meadows, M.E., 1990, Finite Element Simulation of Nonlinear Kinematic Surface Runoff, *Journal of Hydrology,* in press.

Brankensiek, D. L., 1977, Estimating the Effective Capillary Pressure in the Green Ampt Infiltration Equation, *Water Resources Research,* Vol. 13, No. 3, pp. 680-682.

Brankensiek, D. L., and Engleman, R. L., 1979, Relating Initial Abstractions to Soil Infiltration, *SCS-ARS Cooperative Research on Hydrologic Modeling: Progress Report*

Brooks, R. H., and Corey, A. T., 1966, Affecting Fluid Flow, *Journal of the Irrigation and Drainage Division,* ASCE, Vol. 92, No. IR2, pp. 61-68.

Burke, C.B., and Gray, D.D., 1983, Hillslope Runoff Process: A Finite Element Model, Hyd. and Systems Eng. *Rep. No. CE-HSE-83-11,* School of Eng., Purdue Univ.

Chen, C., 1980, Laboratory Verification of a Dam-break Flood Model, *J. Hyd. Div.,* Ross, B.B., Contractor, D.N., and Shanholtz, V.O., 1979, A Finite Element Model of Overland and Channel Flow for Assessing the Hydrologic Impact of Land-use Changes, *J. Hydrol.,* v. 41, pp. 11-30.

Saint-Venant, Barre de, 1871, Theory of Unsteady Water Flow, With Applications to River Floods and to Propagation of Tides in River Channels, *French Academy of Science,* v. 73, p. 148-154, 237-240, Translated into English by Geddings, W.W., Jr., Waterways Experiment Station, No. 49-9, Vicksburg, Miss., U.S.A., 1949.

Stoker, J.J., 1953, Numerical Solution of Flood Prediction and River Regulation Problems, Report I, *Rep. No. IMM-200,* Inst. for Math. and Mech., New York Univ., New York.

Taylor, C., Al-Mashidani, G., and Davis, J.M., 1974, A Finite Element Approach to Watershed Runoff, *J. Hydrol.,* v. 21, pp. 231-246.

Wischmeier, W.H. and Smith, D.D., 1958, Rainfall Energy and Its Relationship to Soil Loss, Transactions at America's Geophysical Union, pp 285 - 291.

Woolhiser, D.A., and Ligget, J.A., 1967, Unsteady One Dimensional Flow over a Plane: The Rising Hydrograph, *Water Resour. Res.,* v. 3, no. 3, pp. 753-771.

Yen, B.C., 1973, Open-channel Flow Equations Revisited, *J. Eng. Mech. Div., Am. Soc. Civ. Eng.,* v. 99, no. EM5, pp. 979-1009.

APPENDIX A: DATA REQUIREMENTS FOR GHIM

As indicated in Chapter I, the geographic hydrologic interface program requires as input four separate data files: NDATA, IDATA, SDATA, and EDATA. All four of these data files are produced by the rill development program RDM. The geographic hydrologic interface program uses the information from these data files to create a single output file (i.e., HDATA) which provides a complete physical characterization of the hydrologic response area. This file is used along with an additional control file (i.e., CDATA) as input to HYDROMIN.

This appendix contains a detailed description of each of the four input data files associated with GHIM. Also included is a description of the required data format associated with each file.

A.1. NDATA

The NDATA file contains information on the number of interrill cells associated with each individual subrillshed along with the number of subrill cells associated with each subrill. The file also contains information on the number of upstream (master) subrills that directly drain into each subrill along with the identification number associated with each of the subrills. The required data format for each line in the NDATA file is shown in Table A.1.

All subrills associated with a particular rill are assigned indentification numbers starting with the exterior subrills first followed by the interior subrills next. The exterior subrill numbers may be assigned in any order while the interior subrills must be numbered sequentially starting with the highest subrill first. An example of this type of numbering scheme is shown in Figure A.1.

Table A.1 Required Data Format for NDATA

Card Group	Format	Columns	Card Description	Variable Name
N1	2x	1-2		
	I5	3-5	Number of Rills	NR
N2	2x	1-2	Card Group Identifier	
	I5	3-5	ID Number of Rill	IDR
	I5	6-10	Number of Subrills per Rill	NSR
N3	2x	1-2	Card Group Identifier	
	I5	3-7	ID Number of Subrill	IDSR
	I5	8-12	Number of Subrill Cells per Subrill	NCSR
	I5	13-17	Number of Interill Cells	NCSA
	5I5	18-22	Number of Masters	ISLVC(I)
		23-42	ID of Each Master	

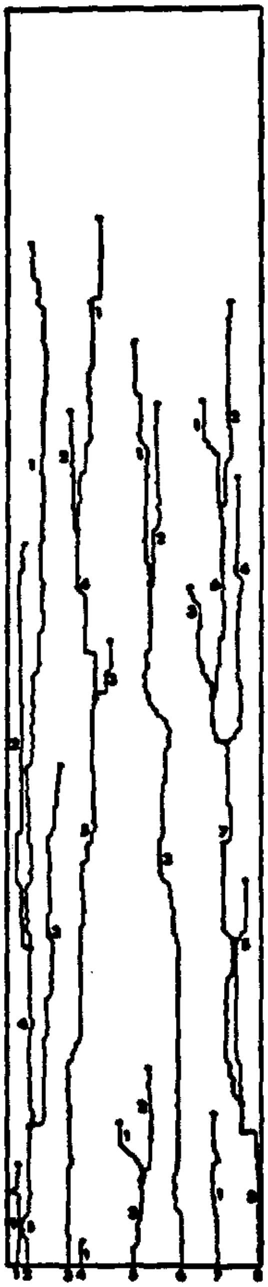

Figure A.1 Rill Network Numbering Scheme

A.2. IDATA

The IDATA file contains the X-Y coordinate pairs for each interrill cell for the entire hydrologic response area. The X-Y coordinate pairs are grouped in the data file according to their associated subrillshed. The blocks of X-Y coordinate pairs are arranged starting with the first subrillshed associated with rill 1 and ending with the last subrillshed associated with rill NR. The number of interrill cells associated with each subrillshed is specified in the NDATA file. The required data format for each line in the IDATA file is shown in Table A.2.

Table A.2 Required Data Format for IDATA

Card Group	Format	Columns	Card Description	Variable Name
I1	2x	1-2	Card Group Identifier	
	I2	3-4	X Coordinate of Area	IXA
	I4	5-8	Y Coordinate of Area	IYA

A.3. SDATA

The SDATA file contains the X-Y coordinate pairs for each subrill cell for the entire hydrologic response area. Associated with each X-Y coordinate pair is the total number of upstream cells which drain into the particular subrill cell. The X-Y coordinate pairs are grouped in the data file according to their associated subrill. The blocks of X-Y coordinate pairs are arranged starting with the first subrill associated with rill 1 and ending with the last subrill associated with rill NR. The number of subrill cells associated with each subrill is specified in the NDATA file. The required data format for each line in the SDATA file is shown in Table A.3.

Table A.3 Required Data Format for SDATA

Card Group	Format	Columns	Card Description	Variable Name
S1	2x	1-2	Card Group Identifier	
	I4	3-6	X Coordinate	IXC
	I6	7-12	Y Coordinate	IYC
	I6	13-18	Number of Area Cells Draining to this Channel	IFQ

A.4. EDATA

The EDATA file contains the elevation associated with each cell in the entire hydrologic response area. The data is grouped in sequential blocks with each block associated with a particular row in the grid network. The required data format for each block of data is shown in Table A.4.

Table A.4 Required Data Format for EDATA

Card Group	Format	Columns	Card Description	Variable Name
E1	2x	1-2	Card Group Identifier	
	I2	3-4	First Elevation Value	IE1(I)
	180I4	5-724	Elevation Values	

APPENDIX B: DATA REQUIREMENTS FOR HYMODRIN

As indicated in Chapter III, HYDROMIN requires two input data files: HDATA and CDATA. The HDATA file is produced by GHIM while CDATA must be produced by the user. Both data files are discussed in detail in the following sections.

B.1. HDATA

The HDATA file contains information regarding the physical characteristics of the hydrologic response area. The HDATA file is divided into five different types of records, or card groups. The first card (i.e., H1) contains the total number of rills in the hydrologic response area. The second card, (i.e., H2) contains the ID number of a particular rill and the associated number of subrills. A separate H2 card is required for each rill in the hydrologic response area. This card must be followed by a separate H3 and H4 card for each subrill along with an associated set of H5 cards. The H3 and H4 cards are used to describe the characteristics of each subrill while the H5 cards are used to describe the charactersitics of the associated subrillshed. The required data format for each HDATA file is provided in Table B.1. An example HDATA file for the rill network shown in Figure A.1. is provided in Table B.2.

As discussed in Chapter II, each subrillshed may be subdivided into a series of interrill flow elements and an additional single source area flow element. Each interrill flow element may be further subdivided into a left plane and a right hand plane. Each H5 card is used to describe the characterisitcs of either a source area flow element, a right hand interrill flow element, or a left hand interrill flow plane. The maximum total number of H5 cards associated with a particular subrill will be equal to $(NIRE_{ij}*2)+1$, where $NIRE_{ij}$ equals the number of interrill elements associated with rill i and subrill j.

Table B.1 Required Data Format for HDATA

Card Group	Format	Columns	Card Description	Variable Name
H1	2x	1-2	Card Group Identifier	
	I3	3-5	Number of Rills	NR
H2	2x	1-2	Card Group Identifier	
	I3	3-5	ID Number of Rill	IDR
	I5	6-10	Number of Subrills per Rill	NUMSC
H3	2x	1-2	Card Group Identifier	
	I8	3-10	ID Number of Rill	IDR
	I10	11-20	ID Number of Subrill	IDSR
	I10	21-30	Number of Channel Elements	NELM
	F10.2	31-40	Channel Length (m)	CLGH
	F10.2	41-50	Channel Slope (mm)	CSLP
	F10.2	51-60	Cummulative Flow Area (m^2)	FCAR
	F10.2	61-70	Channel Width (m)	CHW
	F5.2	71-75	Width Design Flow (cfs)	QWD
	F5.2	76-80	Channel Base Flow (cfs)	QIN
H4	2x	1-2	Card Group Identifier	
	5I5	1-5	Number of Masters	NSLVC(I)
		6-25	ID of Each Master	

Table B.1 Required Data Format for HDATA

Card Group	Format	Columns	Card Description	Variable Name
H5	2x	1-2	Card Group Identifier	
	I5	3-5	ID Number of Subrill Element	IEL
	I5	6-10	ID Number of Plane: 0 = Upper 1 = Right 2 = Left	IP
	I5	11-15	X Coordinate	IX
	I5	16-20	Y Coordinate	IY
	F10.2	21-30	Area of Plane (m^2)	AREA
	F10.2	31-40	Plane Length (m)	PLNG
	F10.2	41-50	Plane Slope (m/m)	PSLP
	F10.2	51-60	Plane Roughness (mm)	PRGH
	F10.2	61-70	Cummulative Flow Area (m^2)	FAVL
	F10.2	71-80	Element Length (m)	ELNG

Table B.2 Example HDATA file

```
H1   9
H2   1   9
H3       1        1        5  11.78560   0.08045   4.48515
H4   0   0   0    0  0
H5   1   1  30  296  0.15063   0.05927   0.17109  10.58602   4.48515   2.54000
H5   1   2  29  394  0.17002   0.04156   0.14640  10.23766   4.16451   2.54000
H5   2   1  30  395  0.12855   0.05054   0.18167   5.01476   4.16322   2.54000
H5   2   2  24  492  0.38564   0.12544   0.09253   9.08366   3.65031   2.54000
H5   3   1  24  493  0.17980   0.07235   0.09804   4.14928   3.64967   2.54000
H5   3   2  26  591  0.16471   0.04156   0.15323   7.75602   3.30580   2.54000
H5   4   1  26  592  0.22478   0.07005   0.11050   2.86814   3.30451   2.54000
H5   4   2  22  686  0.34167   0.08315   0.07979   3.92599   2.73935   2.54000
H5   5   1  22  687  0.16765   0.04789   0.11032   3.07073   2.73870   1.62560
H5   5   2   9  747  1.14977   0.30272   0.09522   6.01680   1.42193   1.62560
H5   6   0   9  747  1.42193   3.14960   0.06865  12.00394   1.42193   0.00000
H3       1        2        2   2.87020   0.09135   3.02451
H4   0   0   0    0  0
H5   1   1  30  296  0.48466   0.10211   0.32534  17.08169   3.02451   2.54000
H5   1   2  38  394  1.20437   0.22834   0.13204  18.91173   1.33548   2.54000
H5   2   1  36  395  0.00402   0.05292   0.05633   1.83722   1.33096   0.33020
H5   2   2  36  406  0.04050   0.52493   0.10382  13.21119   1.29096   0.33020
H5   3   0  36  406  1.29096   3.08779   0.07881   9.11308   1.29096   0.00000
H3       1        3        2   3.17500   0.08911   1.80774
H4   0   0   0    0  0
H5   1   1  25   99  0.21744   0.07061   0.17144   7.03517   1.80774   2.54000
H5   1   2  25  198  0.18772   0.03556   0.29095   6.84882   1.40258   2.54000
H5   2   1  25  199  0.05979   0.13818   0.31095   6.05172   1.40193   0.63500
H5   2   2  27  223  0.04924   0.08839   0.15705   2.95836   1.29355   0.63500
H5   3   0  27  223  1.29355   5.12656   0.10004  16.40262   1.29355   0.00000
H3       1        4        6  13.00480   0.09165   6.51998
H4   0   0   0    0  0
H5   1   1  19   98  0.27016   0.10414   0.20234   7.33167   6.51998   2.54000
H5   1   2  17  197  0.28597   0.08484   0.14314   6.19629   5.96386   2.54000
H5   2   1  17  198  0.28823   0.10730   0.15964   6.50636   5.95870   2.54000
H5   2   2   8  295  0.43371   0.13659   0.10018   6.87874   5.24192   2.54000
H5   3   1   8  296  1.09234   0.21590   0.08861   4.14036   5.23805   2.54000
H5   3   2  13  395  0.65540   0.10414   0.08208   7.39363   3.49419   2.54000
H5   4   1  13  396  0.43567   0.31370   0.09594   6.76520   3.49225   2.54000
H5   4   2  11  492  0.18239   0.10605   0.10437   4.81643   2.87612   2.54000
H5   5   1  11  493  0.74174   0.26365   0.08671   5.91189   2.87548   2.54000
H5   5   2  11  592  0.79890   0.25857   0.07229   7.32300   1.33548   2.54000
H5   6   1  11  593  0.01866   0.25188   0.06282   4.33356   1.33419   0.30480
H5   6   2  11  604  0.02650   0.33232   0.07857   3.44384   1.29032   0.30480
H5   7   0  11  604  1.29032   2.68556   0.08270   4.07609   1.29032   0.00000
```

B.2. CDATA

The CDATA file contains the values of the control parameters for HYDROMIN as well as soil and rainfall data for a particular analysis. The CDATA file contains nine different card groups or lines of data. The required data formats for the various cards in the CDATA file are shown in Table B.3.

The first card of the CDATA file (i.e., C1) contains two lines of title information for use in describing the particular analysis. The C2 card contains the values of several time and iteration limit parameters. These include the total number of time steps (NTS), the time step flow depth profile output number (NFDC), the maximum number of diffusion water iterations (MDWIT), and the maximum number of downstream boundary condition iterations (MBCIT).

The C3 card contains additional numerical analysis parameters. These include the finite difference time weighthing coefficient (RTH), the relative diffusion wave tolerance (DWT), the boundary condition tolerance (BCT) the computational time interval (DT), the tractive force coefficient (TRACF), and the Manning's roughness coefficient (RMANN).

The C4 card contains the time parameters for the input rainfall distribution. These parameters include the total number of time intervals (NRAIN), and the time in seconds associated with each interval (DELTA). The C5 cards contain the rainfall values (cm) associated with each of the time intervals.

The C6 card contains the depth and soil composition of the first soil layer. These parameters include the soil depth (i.e., SL1), the initial soil moisture (SMCI), the percent clay in the soil layer (i.e., CLAY), and the percent sand in the soil layer (i.e., SAND). The C7 card contains the infiltration parameters for the two layer Green Ampt model. These parameters include the wetting front suction head for each soil layer (i.e., SAV1 and SAV2), the total porosity for each soil layer (i.e., POR1 and POR2), and the hydraulic conductivity for each soil layer (i.e., FK1 and FK2).

Table B.3 Required Data Format for CDATA

Card Group	Format	Columns	Card Description	Variable Name
C1	2x	1-2	Card Group Identifier	
	A70	3-72	Title	ITLE
C2	2x	1-2	Card Group Identifier	
	I3	3-5	Number of Time Steps	NTS
	I5	6-10	Time Step Flow Depth Profile Output Number	NFDC
	I5	11-15	Maximum Number of Diffusion Wave Iterations	MDWIT
	I5	15-20	Maximum Number of Downstream Boundary Condition Iterations	MBCIt
C3	2x	1-2	Card Group Identifier	
	F8.2	3-10	Time Weighting Coefficient Between .5 and 1.00	RTH
	F10.2	11-20	Relative Diffusion Wave Tolerance	DWT
	F10.2	21-30	Relative Boundary Condition Tolerance	BCT
	F10.2	31-40	Time Steps in Seconds	DT
	F10.2	41-50	Critical Tractive Force of Soil	TRACF
	F10.2	51-60	Mannings Roughness Coefficient	RMANN

Table B.3 (Cont) Required Data Format for CDATA

Card Group	Format	Columns	Card Description	Variable Name
C4	2x	1-2	Card Group Identifier	
	I8	3-10	Intervals of Time	NRAIN
	F10.1	11-20	Delta	DELTA
C5	2x	1-2	Card Group Identifier	
	F8.0	3-10	Rainfall Read 1 to NRAIN	HY(I)
C6	2x	1-2	Card Group Identifer	
	F8.2	3-10	Soil Depth	SL1
	F10.2	11-20	Initial Soil Moisture	SMCI
	F10.2	21-30	Clay % in Soil Layer	CLAY
	F10.2	31-40	Sand % in Soil Layer	SAND

Table B.3 (Cont) Required Data Format for CDATA

Card Group	Format	Columns	Card Description	Variable Name
C7	2x	1-2	Card Group Identifier	
	F8.2	3-10	Wetting Front Suction Head for Upper Layer	SAV1
	F10.2	11-20	Wetting Front Suction Head for Lower Layer	SAV2
	F10.2	21-30	Total Porosity for Upper Layer	POR1
	F10.2	31-40	Total Porosity for Lower Layer	POR2
	F10.2	41-50	Hydraulic Conductivity for Upper Layer	FK1
	F10.2	51-60	Hydraulic Conductivity for Lower Layer	FK2

APPENDIX C

COMPUTER LISTING OF GHIM

```
C     FILE: GHIM.FORTRAN             LAST REVISION: SEPTEMBER 20, 1990
C
C     PROGRAM TO CREATE INPUT DATA FILE FOR HYDROMIN.FORTRAN AND
C     DYRT.FORTRAN USING OUTPUT DATA FILES FROM GIS.FORTRAN
C     PREPROCESSOR PROGRAM FOR DYNAMIC RILL NETWORK PROJECT USGS
C
C     NOTE: ALL UNIT IN M OR M^2, ACCEPT RANDOM ROUGHNESS IN MM
C
C     WRITTEN BY: LINDELL ORMSBEE  UNIVERSITY OF KENTUCKY, LEXINGTON, KY
C                 CIVIL ENGINEERING DEPARTMENT
C
C     REVISED BY: DANIEL E. STORM  UNIVERSITY OF KENTUCKY, LEXINGTON, KY
C                 AGRICULTURAL ENGINEERING DEPARTMENT
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
C
      COMMON/CDATA/CLNG,CSLP,FCAR,UNITA,UNITL
      COMMON/FILES/IF1,IF2,IF3,IF4,IF5,IF6
      COMMON/IACELL1/IXA(32000)
      COMMON/IACELL2/IYA(32000)
      COMMON/IACELL3/IXLMX(32000),IXRMX(32000)
      COMMON/IBCELL/IXL(32000),IYL(32000),IXR(32000),IYR(32000)
      COMMON/ICCELL/IXC(32000),IYC(32000),IFQ(32000),IFQQ(32000)
      COMMON/IDELL1/IXXA(32000),IYYA(32000)
      COMMON/IDELL2/IXMA(32000),IXCA(32000),IYCA(32000)
      COMMON/INDEX/KNCRA(50,50),KNCAA(50,50)
      COMMON/NDATA/NCSAA(50,50),NCSRA(50,50),NSRA(50)
      COMMON/PDATA/IXM,IYM,IXN,IYN,AREA,PLNG,PSLP,PRGH
      COMMON/IDATA/ISLVC(5),NSLVC(50,50,5)
      COMMON/MXMY/ MX,MY
      COMMON/ELEV/ X(32000),Y(32000),Z(32000),NELEV(181,871)
C
      INTEGER*4 ICN(7),XX(181),BND(16)
      CHARACTER*1  IDISK,EDISK,ODISK
      CHARACTER*5  RCODE
      CHARACTER*25 FNET,FLEV,FBND,NDATA,SDATA,IDATA,EDATA,HDATA
C
C     DEFINE INPUT/OUTPUT DISKS
C
      IDISK='B'
      EDISK='B'
      ODISK='A'
C
C     DEFINE INPUT AND OUTPUT FORMAT STATEMENTS
C
1001  FORMAT(6I6,7I4)
1002  FORMAT(12I6)
1003  FORMAT(3I6)
1004  FORMAT(2I6)
C
2001  FORMAT(4I5,6F10.5)
2002  FORMAT(3I10,4F10.5)
```

```
2003  FORMAT(I5)
2004  FORMAT(2I5)
2005  FORMAT(5I5)
C
C     DEFINE OUTPUT INDEX
C
      IF1 = 1
      IF2 = 2
      IF3 = 3
      IF4 = 0
      IF5 = 7
      IF6 = 0
      IF7 = 9
      IF8 = 10
      IF9 = 11
C
C     INPUT RUN CODE
C
      WRITE(6,*) 'INPUT RUN CODE (EX: S2R2)'
      READ (6,1) RCODE
    1 FORMAT(A5)
C     RCODE='S2R2'
C
C     DEFINE FILE NAMES
C
      FBND ='/'//RCODE//' BND '//IDISK
      FLEV ='/'//RCODE//' LEV '//EDISK
      FNET ='/'//RCODE//' NET '//IDISK
C
      NDATA='/'//RCODE//' CHN '//IDISK
      SDATA='/'//RCODE//' SDATA '//IDISK
      IDATA='/'//RCODE//' SHD   '//IDISK
      HDATA='/'//RCODE//' HDA   '//ODISK
C
C......................................................................
C                                                                     .
C     SET CONTROL PARAMETERS                                          .
C                                                                     .
C     MX    = MAXIMUM NUMBER OF ROWS OF ELEVATION DATA                .
C     MY    = MAXIMUM NUMBER OF COLUMNS OF ELEVATION DATA             .
C     NCPE  = NUMBER OF CELLS  PER  ELEMENT                           .
C     UNITA = UNIT AREA OF EACH CELL    (M)                          .
C     UNITL = UNIT LENGTH OF EACH CELL (M)                           .
C
C     MINE  = MINIMUM NUMBER OF ELEMENTS PER SUBRILL
C     MAXE  = MAXIMUM NUMBER OF ELEMENTS PER SUBRILL
C                                                                     .
C......................................................................
C
      UNITA = 1.0/144.*0.3048**2
      UNITL = 1.0/12.0*0.3048
      MX=181
      MY=871
```

```
C
      NCPE1=12
      MINE=2
      MAXE=71
C
C     OPEN INPUT/OUTPUT DATA FILES
C
      OPEN(IF1,FILE=NDATA)
      CALL FILEINF(IRCODE,'RECFM','F','MAXREC',10000,'LRECL',4*MX)
      OPEN(IF2,FILE=SDATA,ACCESS='DIRECT',FORM='UNFORMATTED',RECL=18   )
      CALL FILEINF(IRCODE,'RECFM','F','MAXREC',MX*MY,'LRECL',4*MX)
      OPEN(IF3,FILE=IDATA,ACCESS='DIRECT',FORM='UNFORMATTED',RECL=8    )
      OPEN(IF5,FILE=HDATA)
C
C     READ IN DATA FILES FROM GIS.FORTRAN AND CONVERT TO DIRECT ACCESS
C
      OPEN(IF7,FILE=FNET)
      I=0
    5 READ(IF7,6,END=7) I1,I2,I3
    6 FORMAT(3I6)
      I=I+1
      WRITE(IF2,REC=I) I1,I2,I3
      GOTO 5
    7 CLOSE (IF7)
C
C     READ IN ELEVATION DATA
C
      MX2=MX+2
      CALL FILEINF(IRCODE,'RECFM','F','MAXREC',MY+2,'LRECL',4*MX2)
      OPEN(IF8,FILE=FLEV,ACCESS='DIRECT',FORM='UNFORMATTED',RECL=MX2*4)
      DO 8 I=1,MY
         K=MY-I+1
         READ(IF8,REC=I+1) L,(NELEV(J,K),J=1,MX)
    8 CONTINUE
      CLOSE (IF8)
C
C.....................................................................
C                                                                     .
C     READ IN CONTROL DATA                                            .
C                                                                     .
C     NR   = TOTAL NUMBER OF RILLS IN DATA SET                        .
C     NS   = TOTAL NUMBER OF SUBRILLSHEDS IN THE DATA SET             .
C                                                                     .
C     IDR  = RILL ID NUMBER                                           .
C     NSR  = NUMBER OF SUBRILLS ASSOCIATED WITH RILL IDR              .
C     NCSR = NUMBER OF CELLS ASSOCIATED WITH SUBRILL IDSR             .
C     IDSR = SUBRILL ID NUMBER                                        .
C     NCSB = NUMBER OF CELLS WHICH DEFINE THE SUBRILLSHED BOUNDARY    .
C            ASSOCIATED WITH SUBRILL IDSR                             .
C     NCSA = NUMBER OF CELLS WHICH DEFINE THE SUBRILLSHED ASSOCIATED  .
C            WITH SUBRILL IDSR                                        .
C                                                                     .
C.....................................................................
```

```
C
      OPEN(IF9,FILE=FBND)
      READ(IF1,1001)NR,NS
C
C     CHECK FOR ARRAY EXCEEDING DIMENSIONS
C
      IF(NR.GT.49) THEN
         WRITE(*,*) 'NR NUMBER OF RILLS EXCEEDED',IDR
         STOP
      ENDIF
C
      KNCR=1
      KNCA=1
C
      I=0
    9 I=I+1
      IF(I.GT.NS) GOTO 19
      READ(IF1,1001)IDR,NSR,NCSR,IDSR,NCSB,NCSA,(ICN(J),J=1,7)
C
C     CHECK FOR ARRAY EXCEEDING DIMENSIONS
C
      IF(NCSA.GT.32000) THEN
         WRITE(*,*) 'NCSA NUMBER OF SUBRILL CELLS EXCEEDED',NCSA
         STOP
      ENDIF
      IF(NCSR.GT.32000) THEN
         WRITE(*,*) 'NCSR NUMBER OF SUBRILL CELLS EXCEEDED',NCSR
         STOP
      ENDIF
      IF(NSR.GT.50) THEN
         WRITE(*,*) 'NSR NUMBER OF SUBRILLS EXCEEDED',NSR
         STOP
      ENDIF
C
C     CHECK FOR NULL SUBSHED
C
      IF(IDR.EQ.0) THEN
         NS=I
         NR=NR+1
         IDR=NR
      ENDIF
C
C     COUNT NUMBER OF UNSTREAM BRANCHES
C
      ICNT=0
      DO 16 J=1,7
         IF(ICN(J).EQ.0) GOTO 17
         ICNT=ICNT+1
   16 CONTINUE
   17 ISLVC(1)=ICNT
      ISLVC(2)=ICN(1)
      ISLVC(3)=ICN(2)
      ISLVC(4)=ICN(3)
```

```
      ISLVC(5)=ICN(4)
C
      IF(ICNT.GT.4) THEN
         WRITE(6,*) 'NUMBER OF INCOMING BRANCHES EXCEEDED'
         STOP
      ENDIF
C
      NSRA(IDR)=NSR
      NCSAA(IDR,IDSR)=NCSA
      NCSRA(IDR,IDSR)=NCSR
C
      DO 10 J=1,5
      NSLVC(IDR,IDSR,J)=ISLVC(J)
10    CONTINUE
C
C     ASSIGN CUMMULATIVE SCRATCH FILE INDEX FOR EACH SUBRILL
C
      KNCRA(IDR,IDSR)=KNCR
      KNCAA(IDR,IDSR)=KNCA
C
C     UPDATE INDEX
C
      KNCR=KNCR+NCSR
      KNCA=KNCA+NCSA
C
      GOTO 9
   19 CLOSE (IF9)
C
C.......................................................................
C                                                                      .
C     LOOP OVER RILLS                                                  .
C                                                                      .
C.......................................................................
C
      WRITE(IF5,2003)NR
      NRSTP=NR-1
C
      DO 170 IDR=1,NRSTP
      WRITE(IF5,2004)IDR,NSRA(IDR)
C
C.......................................................................
C                                                                      .
C     LOOP OVER SUB RILLLS  (IDSR = SUB RILL ID NUMBER)                .
C                                                                      .
C.......................................................................
C
        DO 160 IDSR=1,NSRA(IDR)
           WRITE(*,33) 'PROCESSING RILL # =',IDR,' SUBRILL # =',IDSR,
     *                 ' OF ',NSRA(IDR)
   33      FORMAT(A20,I3,A13,I3,A4,I3)
C
           ITEST=NSLVC(IDR,IDSR,1)
           NCSR=NCSRA(IDR,IDSR)
```

```
          KONE=KNCRA(IDR,IDSR)
          KEND=KONE+NCSR-1
C
C......................................................................
C                                                                     .
C     READ IN CELLS FOR SUBRILL IDSR                                  .
C                                                                     .
C     NCSR   = NUMBER OF CELLS WHICH MAKE UP SUBRILL IDSR              .
C     IXC(K) = X COORDINATE OF CELL K                                 .
C     IYC(K) = Y COORDINATE OF CELL K                                 .
C     IFQ(K) = FLOW ACCUMULATION UNITS FOR CELL A (I.E. THE TOTAL     .
C               NUMBER OF CELLS THAT DRAIN TO THIS CELL)              .
C                                                                     .
C......................................................................
C
           ICE = 0
           DO 20 K=KONE,KEND
              ICE=ICE+1
              READ(IF2,REC=K)IXC(ICE),IYC(ICE),IFQ(ICE)
C     WRITE(*,*) K,IFQ(ICE)
20         CONTINUE
C
C......................................................................
C                                                                     .
C     DETERMINE CHARACTERISTICS OF SUBRILL IDSR                       .
C                                                                     .
C......................................................................
C
           CALL SUBRC(NCSR,IDSR,NSRA(IDR))
C
C......................................................................
C                                                                     .
C     DETERMINE TOTAL NUMBER OF ELEMENTS PER SUBRILL                  .
C                                                                     .
C     NCSR = NUMBER OF CELLS PER SUBRILL                              .
C     NCPE = NUMBER OF CELLS PER ELEMENT                              .
C     NESR = NUMBER OF ELEMENTS PER SUBRILL                           .
C                                                                     .
C     CLNG = SUBRILL LENGTH                                           .
C     ELNG = ELEMENTAL SUBRILL LENGHT                                 .
C     PLNG = INTERRILL PLANE LENGTH                                   .
C                                                                     .
C     MINE  = MINIMUM NUMBER OF ELEMENTS PER SUBRILL                  .
C     MAXE  = MAXIMUM NUMBER OF ELEMENTS PER SUBRILL                  .
C                                                                     .
C......................................................................
C
           NCPE=NCPE1
           NESR=NCSR/NCPE+1
           IF(FLOAT(NCSR)/FLOAT(NCPE)+1..EQ.FLOAT(NESR)) NESR=NESR-1
           IF(NESR.LE.MINE-1.OR.NCSR.EQ.NCPE) NCPE=(NCSR+1)/MINE
           IF(NESR.GE.MAXE) NCPE=(NCSR/MAXE)+1
           IF(NESR.LE.MINE-1.OR.NCSR.EQ.NCPE) NESR=MINE
```

```
            IF(NESR.GE.MAXE) THEN
               NESR=NCSR/NCPE+1
               IF(FLOAT(NCSR)/FLOAT(NCPE)+1..EQ.FLOAT(NESR)) NESR=NESR-1
            ENDIF
C
            NEL=NESR
      WRITE(*,*) 'NEL=',NEL
C
            WRITE(IF5,2002)IDR,IDSR,NEL,CLNG,CSLP,FCAR
            WRITE(IF5,2005)(NSLVC(IDR,IDSR,ID),ID=1,5)
C
C.....................................................................
C                                                                    .
C     READ IN AREA CELLS FOR SUBRILLSHED IDSR                        .
C                                                                    .
C     NCSA   = NUMBER OF CELLS IN SUBRILLSHED IDSR                   .
C     IXA(K) = X COORDINATE FOR CELL K                               .
C     IYA(K) = Y COORDINATE FOR CELL K                               .
C                                                                    .
C.....................................................................
C
            NCSA=NCSAA(IDR,IDSR)
            KONE=KNCAA(IDR,IDSR)
            KEND=KONE+NCSA-1
C
            ICE=0
            DO 30 K=KONE,KEND
               ICE=ICE+1
               READ(IF3,REC=K)IXA(ICE),IYA(ICE)
30          CONTINUE
C
C.....................................................................
C                                                                    .
C     INIALIZE COUNTERS                                              .
C                                                                    .
C     NEC  = NUMBER OF SUBRILL CELLS IN ELEMENT ISRE                 .
C     NLC  = NUMBER OF AREA CELLS IN LEFT PLANE                      .
C     NRC  = NUMBER OF AREA CELLS IN RIGHT PLANE                     .
C     ISRE = INDEX OF SUBRILL ELEMENT                                .
C.....................................................................
C
            NEC2=0
            NEC=0
            NLC=0
            NRC=0
            ISRE=1
C
C.....................................................................
C                                                                    .
C     LOOP OVER EACH SUBRILL CELL                                    .
C                                                                    .
C.....................................................................
C
```

```
            DO 120 K1=1,NCSR
               NEC2=NEC2+1
C
C           IF TWO SUBRILL CELLS ARE SIDE BY SIDE, SKIP TO NEXT CELL
C
               IF(NEC.GT.0) THEN
                  IF(IYCA(NEC).EQ.IYC(K1)) THEN
                     IXCA(NEC)=IXC(K1)
                     IXLMX(NEC)=IXC(K1)
                     IXRMX(NEC)=IXC(K1)
                     IFQQ(NEC)=IFQ(K1)
                     GOTO 65
                  ENDIF
               ENDIF
C
               NEC=NEC+1
               IXCA(NEC)=IXC(K1)
               IYCA(NEC)=IYC(K1)
               IXLMX(NEC)=IXC(K1)
               IXRMX(NEC)=IXC(K1)
               IFQQ(NEC)=IFQ(K1)
C
C......................................................................
C                                                                     .
C     LOOP OVER AREA CELLS (I.E. FIND AREA CELLS ON THE SAME ROW AS   .
C                           THE CURRENT SUBRILL CELL)                  .
C                                                                     .
C                           DETERMINE OUTER LEFT AND RIGHT CELL ON     .
C                           EACH ROW (I.E. IXLMX AND IXRMX)            .
C                                                                     .
C......................................................................
C
               IXCZ=IXC(K1)
               IYCZ=IYC(K1)
C
               DO 60 K2=1,NCSA
                  IF(IYA(K2).EQ.IYCZ) THEN
                     IF(IXA(K2).LT.IXCZ) THEN
                        NLC=NLC+1
                        IXL(NLC)=IXA(K2)
                        IYL(NLC)=IYCZ
                        IF(IXL(NLC).LT.IXLMX(NEC))IXLMX(NEC)=IXL(NLC)
                     ELSE
                        NRC=NRC+1
                        IXR(NRC)=IXA(K2)
                        IYR(NRC)=IYCZ
                        IF(IXR(NRC).GT.IXRMX(NEC))IXRMX(NEC)=IXR(NRC)
                     ENDIF
                  ENDIF
   60          CONTINUE
C
   65           IF(K1.EQ.NCSR)GO TO 70
                IF(NEC2.LT.NCPE)GO TO 120
```

```
  70        CONTINUE
C
C.......................................................................
C                                                                      .
C     DETERMINE CHARACTERISTICS OF EACH PAIR OF ELEMENT PLANES         .
C                                                                      .
C.......................................................................
C
C     DEFINE FLOW AREAS FOR A SUBSHED ELEMENT
C
            ELNG=DFLOAT(NEC2)*UNITL
C
            AREAL=DFLOAT(NLC)*UNITA
            AREAR=DFLOAT(NRC)*UNITA
C
            IF(ISRE.EQ.1) THEN
               FAVL=DFLOAT(IFQ(2))*UNITA
            ELSE
               FAVL=DFLOAT(IFQQ(1))*UNITA
            ENDIF
C
            IF(ISRE.EQ.NESR.AND.ITEST.GT.0) THEN
               FAVR=DFLOAT(IFQQ(NEC-1))*UNITA
            ELSE
               FAVR=DFLOAT(IFQQ(NEC))*UNITA
            ENDIF
C
            IF(ISRE.EQ.1) FOLD=FAVL
            AT=AREAL+AREAR
            DIFF=0.
            IF(AT.GT.0.) DIFF=(FOLD-FAVR-AT)/AT
            AREAL=AREAL*(1.+DIFF)
            AREAR=AREAR*(1.+DIFF)
C
            IF(NEC.EQ.1) THEN
               AREAL=UNITA
               AREAR=UNITA
               FAVL=DFLOAT(IFQQ(NEC))*UNITA
               FAVR=FAVL
            ENDIF
            FOLD=FAVR
C
C     DETERMINE LEFT ELEMENT CHARACTERISTICS
C
            DO 80 K2=1,NLC
               IXXA(K2)=IXL(K2)
               IYYA(K2)=IYL(K2)
80          CONTINUE
C
            DO 90 K2=1,NEC
               IXMA(K2)=IXLMX(K2)
90          CONTINUE
C
```

```
                IP=1
                IFLG=1
                CALL SPLANE (NLC,NEC,IFLG,IP)
C
       WRITE(IF5,2001)ISRE,IP,IXCA(1),IYCA(1),AREAL,PLNG,PSLP,
     1                PRGH,FAVL,ELNG
C
C     DETERMINE RIGHT ELEMENT CHARACTERISTICS
C
                DO 100 K2=1,NRC
                   IXXA(K2)=IXR(K2)
                   IYYA(K2)=IYR(K2)
100             CONTINUE
C
                DO 110 K2=1,NEC
                   IXMA(K2)=IXRMX(K2)
110             CONTINUE
C
                IP=2
                IFLG=1
                CALL SPLANE (NRC,NEC,IFLG,IP)
C
                M=NEC
      WRITE(IF5,2001) ISRE,IP,IXCA(M),IYCA(M),AREAR,PLNG,PSLP,
     1                PRGH,FAVR,ELNG
C
C     RESET COUNTERS
C
                NLC=0
                NRC=0
                NEC=0
                NEC2=0
                ISRE = ISRE+1
C
120         CONTINUE
C
C...........................................................
C                                                          .
C     UPPER RILL SUBRILLSHED                               .
C                                                          .
C...........................................................
C
            IF(ITEST.GT.0) GOTO 160
C
            NEC=0
            NAC=0
            IYMIN=IYC(NCSR)
C
C     DETERMINE SUBRILLSHED BOUNDARY POINTS
C
            DO 140 K1=1,MX
C
               IXMAX=IYMIN
```

```
C
                DO 130 K2=1,NCSA
                   IF(IXA(K2).NE.K1.OR.IYA(K2).LE.IYMIN)GO TO 130
                   NAC=NAC+1
                   IXXA(NAC)=IXA(K2)
                   IYYA(NAC)=IYA(K2)
                   IF(IYA(K2).GT.IXMAX)IXMAX=IYA(K2)
C
130             CONTINUE
C
                IF(IXMAX.GT.IYMIN) THEN
                   NEC=NEC+1
                   IYCA(NEC)=K1
                   IXCA(NEC)=IYMIN
                   IXMA(NEC)=IXMAX
                ENDIF
C
140          CONTINUE
C
C     GENERATE CHARACTERISTICS OF RILLSHED
C
             IF(NEC.LE.0) THEN
                WRITE(*,*) 'NEC LESS THAN 0'
                STOP
             ENDIF
C
             IP=0
             IFLG=2
             CALL SPLANE(NAC,NEC,IFLG,IP)
C
             IDM1=IXC(NCSR)
             IDM2=IYC(NCSR)
             AREA=DFLOAT(IFQ(NCSR))*UNITA
             FAV=DFLOAT(IFQ(NCSR))*UNITA
             ELNG=0.
C
            WRITE(IF5,2001)ISRE,IP,IDM1,IDM2,AREA,PLNG,PSLP,
     *                     PRGH,FAV,ELNG
C
160   CONTINUE
170   CONTINUE
C
C
C......................................................................
C                                                                     .
C     NULL RILL SUBRILLSHED                                           .
C                                                                     .
C......................................................................
C
      IP=0
      NEL=0
      ISRE=0
      IDR=NR
```

```
      IDSR=1
C
      CLNG=0.
      CSLP=0.
      FCAR=0.
      ELNG=0.
C
      WRITE(IF5,2004)IDR,IDSR
      WRITE(IF5,2002)IDR,IDSR,NEL,CLNG,CSLP,FCAR
      WRITE(IF5,2005)(NSLVC(IDR,IDSR,ID),ID=1,5)
C
C     READ IN CELL DATA COORDINANTS FOR NULL RILL SUBRILLSHED
C
          NCSA=NCSAA(IDR,1)
          KONE=KNCAA(IDR,1)
          KEND=KONE+NCSA-1
C
          NAC=0
C
          DO 180 K=KONE,KEND
            NAC=NAC+1
            READ(IF3,REC=K)IXA(NAC),IYA(NAC)
            IXXA(NAC)=IXA(NAC)
            IYYA(NAC)=IYA(NAC)
180       CONTINUE
C
          IYMIN=1
          NEC=0
C
          DO 200 K1=1,MX
C
             IXMAX=IYMIN
C
             DO 190 K2=1,NCSA
                IF(IXA(K2).NE.K1)GO TO 190
                IF(IYA(K2).GT.IXMAX)IXMAX=IYA(K2)
190          CONTINUE
C
             IF(IXMAX.GT.IYMIN) THEN
                NEC=NEC+1
                IYCA(NEC)=K1
                IXCA(NEC)=IYMIN
                IXMA(NEC)=IXMAX
             ENDIF
C
200       CONTINUE
C
C     GENERATE CHARACTERISTICS OF RILLSHED
C
          IP=0
          IFLG=2
          CALL SPLANE(NAC,NEC,IFLG,IP)
C
```

```
            IDM1=0
            IDM2=0
            AREA=DFLOAT(NCSA)*UNITA
            FAV=DFLOAT(NCSA)*UNITA
C
           WRITE(IF5,2001)ISRE,IP,IDM1,IDM2,AREA,PLNG,PSLP,
     *                    PRGH,FAV,ELNG
C
C
      CLOSE (IF1)
      CLOSE (IF2)
      CLOSE (IF3)
      CLOSE (IF5)
C
      END
C.....................................................................
C                                                                     .
C       THIS SUBROUTINE DETERMINES THE LENGTH, SLOPE,                 .
C       CUMMULATIVE FLOW AREA FOR AN ENTIRE SUBRILL                   .
C                                                                     .
C.....................................................................
C
      SUBROUTINE SUBRC(NCSR,IC1,IC2)
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
C
      COMMON/CDATA/CLNG,CSLP,FCAR,UNITA,UNITL
      COMMON/FILES/IF1,IF2,IF3,IF4,IF5,IF6
      COMMON/ICCELL/IXC(32000),IYC(32000),IFQ(32000),IFQQ(32000)
      COMMON/MXMY/ MX,MY
      COMMON/ELEV/ X(32000),Y(32000),Z(32000),NELEV(181,871)
C
C     DETERMINE SUBRILL LENGTH
C
      CLNG=DFLOAT(NCSR)*UNITL
C
C     DETERMINE SUBRILL SLOPE    EQ: Z = M * Y + B
C
      SUMY=0.
      SUMZ=0.
      SUMYZ=0.
      SUMYY=0.
      SUMZZ=0.
C
      DO 10 I=1,NCSR
         LX=IXC(I)
         LY=IYC(I)
         Y1=DFLOAT(LY)*UNITL
         Z1=DFLOAT(NELEV(LX,LY))/1000.
C
         SUMY=SUMY+Y1
         SUMZ=SUMZ+Z1
         SUMYZ=SUMYZ+Y1*Z1
```

```
         SUMYY=SUMYY+Y1*Y1
         SUMZZ=SUMZZ+Z1*Z1
   10 CONTINUE
C
      CSR=DFLOAT(NCSR)
      YBAR=SUMY/CSR
      ZBAR=SUMZ/CSR
      SYY=SUMYY-CSR*YBAR*YBAR
      SYZ=SUMYZ-CSR*YBAR*ZBAR
C
      CSLP=DABS(SYZ/SYY)
C
C     DETERMINE CUMMULATIVE FLOW AREA
C
      II=2
      IF(IC1.EQ.IC2) II=1
      FCAR=DFLOAT(IFQ(II))*UNITA
C
      RETURN
      END
C
C
C.......................................................................
C                                                                      .
C       THIS SUBROUTINE DETERMINES THE CHARACTERSITICS OF THE EACH     .
C       ELEMENT SUB PLANE                                              .
C                                                                      .
C       DETERMINED PARAMETERS INCLUDE: PLANE AREA, LENGTH, SLOPE AND   .
C                                      INITIAL RANDOM ROUGHNESS        .
C                                                                      .
C.......................................................................
C
      SUBROUTINE SPLANE(NNN,NCC,IFLG,IP)

      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)

      COMMON/CDATA/CLNG,CSLP,FCAR,UNITA,UNITL
      COMMON/FILES/IF1,IF2,IF3,IF4,IF5,IF6
      COMMON/IDELL1/IXXA(32000),IYYA(32000)
      COMMON/IDELL2/IXMA(32000),IXCA(32000),IYCA(32000)
      COMMON/PDATA/IXM,IYM,IXN,IYN,AREA,PLNG,PSLP,PRGH
      COMMON/MXMY/ MX,MY
      COMMON/ELEV/ X(32000),Y(32000),Z(32000),NELEV(181,871)
C
C     FIND ELEVATIONS
C
C     WRITE(*,*) 'NCC=',NCC,'  NNN=',NNN,IFLG,IP
      DO 10 I=1,NNN
         LX=IXXA(I)
         LY=IYYA(I)
         X(I)=DFLOAT(LX)*UNITL
         Y(I)=DFLOAT(LY)*UNITL
         Z(I)=DFLOAT(NELEV(LX,LY))/1000.
```

```
   10 CONTINUE
C
C     DETERMINE LENGHT OF ELEMENT
C
      NT=0
      DO 20 I=1,NCC
   20 NT=NT+ABS(IXCA(I)-IXMA(I))
      IF(NCC.GT.0) THEN
         PLNG=DFLOAT(NT)*UNITL/DFLOAT(NCC)
      ELSE
         PLNG=UNITL
      ENDIF
C
C     GENERATE SUB PLANE RANDOM ROUGHNESS
C
      CALL ROUGH (NNN,PRGH,PSLP)
C
      RETURN
      END
C
      SUBROUTINE ROUGH (NP,RR,PSLP)
C        .......................................................
C        .                                                     .
C        .   THIS SUBROUTINE FINDS THE BEST FIT PLANE FOR      .
C        .   THE SUBPLANE AND CALCULATES THE STANDARD          .
C        .   DEVIATION OF THE POINTS FROM THE PLANE WHICH      .
C        .   WILL BE ASSUMED EQUAL TO THE INITIAL RANDOM       .
C        .   ROUGHNESS OF THE SUBPLANE.                        .
C        .                                                     .
C        .......................................................
C
        IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
C
      COMMON/FILES/IF1,IF2,IF3,IF4,IF5,IF6
      COMMON/IDELL1/IXXA(32000),IYYA(32000)
      COMMON/IDELL2/IXMA(32000),IXCA(32000),IYCA(32000)
      COMMON/MXMY/ MX,MY
      COMMON/ELEV/ X(32000),Y(32000),Z(32000),NELEV(181,871)
C
C       INITIALIZE VARIABLES.
C
        SUMX=0.
        SUMY=0.
        SUMZ=0.
        SUMXSQ=0.
        SUMYSQ=0.
        SUMXY=0.
        SUMXZ=0.
        SUMYZ=0.
C
        IF(NP.LE.3)GO TO 40
C
C       CALCULATE SUMS NEEDED TO FIND BEST FIT PLANE.
```

```
C
       DO 20 L=1,NP
          X1=X(L)
          Y1=Y(L)
          Z1=Z(L)
          SUMX=SUMX + X1
          SUMY=SUMY + Y1
          SUMZ=SUMZ + Z1
          SUMXSQ=SUMXSQ + X1*X1
          SUMYSQ=SUMYSQ + Y1*Y1
          SUMXY=SUMXY + X1*Y1
          SUMXZ=SUMXZ + X1*Z1
          SUMYZ=SUMYZ + Y1*Z1
20    CONTINUE
C
C     EQUATION FOR PLANE IS:
C     Z = B0 + B1*X + B2*Y
C
      C1=SUMZ
      C2=SUMXZ
      C3=SUMYZ
      A12=SUMX
      A22=SUMXSQ
      A32=SUMXY
      A11=NP
      A21=SUMX
      A31=SUMY
      A13=SUMY
      A23=SUMXY
      A33=SUMYSQ
C
      DET= A11*((A22*A33) - (A23*A32))
     1   - A12*((A21*A33) - (A23*A31))
     2   + A13*((A21*A32) - (A22*A31))
C
      D1 =  C1*((A22*A33) - (A23*A32))
     1   - A12*(( C2*A33) - (A23* C3))
     2   + A13*(( C2*A32) - (A22* C3))
C
      D2 = A11*(( C2*A33) - (A23* C3))
     1   -  C1*((A21*A33) - (A23*A31))
     2   + A13*((A21* C3) - ( C2*A31))
C
      D3 = A11*((A22* C3) - ( C2*A32))
     1   - A12*((A21* C3) - ( C2*A31))
     2   +  C1*((A21*A32) - (A22*A31))
C
      IF(DET.NE.0)GO TO 25
      WRITE(*,101)
 101  FORMAT('  DETERMINATE EQUAL ZERO, DEFAULT ROUGHNESS USED')
      RR=10.
      RETURN
C
```

```
25     B0 = D1/DET
       B1 = D2/DET
       B2 = D3/DET
C
C      DEFINE SLOPE
C
       PSLP=DSQRT(B1**2+B2**2)
C
C      CALCULATE STANDARD DEVIATION OF POINTS ABOUT PLANE.
C      NOTE: THIS IS DEFINED AS THE INITIAL RANDOM ROUGHNESS IN MM.
C
       SR=0
       DO 30 L=1,NP
          PRED= B0 + B1*X(L) + B2*Y(L)
          OBS =Z(L)
          SR=SR + ((PRED-OBS))**2
30     CONTINUE
C
       RR=SQRT(SR/DFLOAT(NP-3))*1000.
       RETURN
C
40     WRITE(*,102) NP
102    FORMAT(' INSUFFICIENT DATA POINTS, DEFAULT ROUGHNESS USED',I5)
       RR=10.
C
       RETURN
       END
```

APPENDIX D

COMPUTER LISTING OF HYMODRIN

```
C
C
C                             PROGRAM HYMODRIN
C
C                                    BY
C
C                            LINDELL E. ORMSBEE
C
C                                    AND
C
C                           GEORGE  E. BLANDFORD
C
C
C                        LATEST REVISION SEPT 26 1990
C
      PROGRAM MAIN
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
C
      COMMON/CDATA/CLGH,CSLP,CWTH,FCAR,RMANN,TRACF,YIN
      COMMON/PDATA/AREA,PLNG,PRGH,PSLP,CLAY,SAND
      COMMON/DFILES/IF1,IF2
      COMMON/EQN/A(25,12),RHS(11,12),JD(11,12)
      COMMON/FILES/IIN1,INN2,IOUT
      COMMON/FLOWD/QL(11,1000)
      COMMON/FLOWA/QO(1000)
      COMMON/FLOWB/QU(5,1000)
      COMMON/FLWMAT/AFM(3,3,2),QFV(3,2),VFM(3,3,2)
      COMMON/ICHDAT/IDWIT(12),NLF(12),NEL(12),NH(12),NNP(12),NSLVC(5,12)
      COMMON/IDATA/IEL,ISC,NEND,NEQ,NTMC,NUMEL,NUMNP,NUMSC
      COMMON/IELMNT/LD(3,10,12),NCON(3,10,12),NELTP(10,12)
      COMMON/ITDAT/ANORM,BCT,DWT,MBCIT,MDWIT,NBCS,NCSC
      COMMON/LFILES/IFL(12)
      COMMON/NDATA/NSRA(20)
      COMMON/FLOW/AFLW(12),DFLW(12),QFLW(12),QH(10)
      COMMON/RAIND/AKE(2000),HY(60),RE(2000),RN(2000),RAV
      COMMON/RCHDAT/CHR(12),CHS(12),CHW(12),CHZ(12),DSQ(12),WS(12)
      COMMON/RDATA/ONE,R23,R35,R53,ZERO
      COMMON/RELMNT/ELN(10,12),QLT1(10,12),QLT2(10,12)
      COMMON/RLOG/AFAC,BACK
      COMMON/RNODE/AT1(25,12),AT2(25,12),FDT(25,12),FRT(25,12),
     1             SF(25,12),XND(25,12)
      COMMON/TIME/DT,RTH,RTH1,RTT,DELTA,IFDC,ITS,NFDC,NRAIN,NTS
C
      LOGICAL AFAC,BACK
C
      CHARACTER*14 IINC,IIND,IOT1
      CHARACTER*72 ITLE(2)
C
C     INITIALIZING THE SOLUTION LOGICAL VARIABLES FOR
C     MATRIX FACTORIZATION.
C
      AFAC = .TRUE.
```

```
      BACK = .FALSE.
C
1001  FORMAT(3I10,5F10.2)
1002  FORMAT(6I5)
1003  FORMAT(4I5,6F10.0)
1004  FORMAT(2I5)
C
C     INITIALIZING THE PROGRAM CONSTANTS.
C
      IFDC = 0
      ONE = 1.D0
      R23 = 2.D0/3.
      R35 = 3.D0/5.
      R53 = 5.D0/3.
      ZERO = 0.D0
      NTMC = 0
C
C     INPUTTING THE INPUT CONTROL FILE NAME.
C
      WRITE(*,'(A)') '   ENTER THE INPUT CONTROL FILE NAME: A14  -  '
      READ(*,'(A)') IINC
C
C     INPUTTING THE INPUT DATA FILE NAME
C
      WRITE(*,'(A)') '   ENTER THE INPUT DATA FILE NAME: A14  -  '
      READ(*,'(A)') IIND
C
C     INPUTTING THE OUTPUT DATA FILE NAME.
C
      WRITE(*,'(A)') '   ENTER THE OUTPUT DATA FILE NAME: A14  -  '
      READ(*,'(A)') IOT1
C
C     OPEN SCRATCH FILES
C
      IF1 = 1
      IF2 = 2
C
      OPEN(IF1,STATUS='SCRATCH',FORM='UNFORMATTED')
      OPEN(IF2,STATUS='SCRATCH',FORM='UNFORMATTED')
C
C     DATA DISK FILE DEFINITIONS.
C
      IIN1 = 3
      IIN2 = 4
      IOUT = 5
C
      OPEN(IIN1,FILE=IINC)
      OPEN(IIN2,FILE=IIND)
      OPEN(IOUT,FILE=IOT1)
C
C     OPEN DEBUG FILE
C
      OPEN(6,STATUS='OLD',FORM='FORMATTED',FILE='DATA6')
```

```
C
C     OPENING THE CHANNEL LATERAL INFLOW SCRATCH DISK FILES.
C
      DO 10 ISC=1,3
         IFL(ISC) = ISC + 6
         OPEN(IFL(ISC),STATUS='SCRATCH',FORM='UNFORMATTED')
C        OPEN(IFL(ISC),STATUS='SCRATCH',ACCESS='DIRECT',RECL=80)
10    CONTINUE
C
C C
..................................................................
C     .                                                                .
C     .    READ IN SYSTEM DATA                                         .
C     .                                                                .
C     ..................................................................
C
      CALL SDATA
C
C     ..................................................................
C     .                                                                .
C     .    BEGIN PROGRAM LOOP                                          .
C     .                                                                .
C     ..................................................................
C
      READ(IIN2,1004)NR,IFLAG
C
      DO 50 IDR=1,NR-1
C
      READ(IIN2,1004)IDUM,NUMSC
C
        NTMC=0
C
        DO 40 ISC=1,NUMSC
C
C     ..................................................................
C     .                                                                .
C     .    BEGIN PROGRAM LOOP                                          .
C     .                                                                .
C     .    NR    - NUMBER OF RILLS                                     .
C     .    IFLAG - (IFLAG = 0) - CUMMULATIVE FLOW AREA USED FOR        .
C     .                          DETERMINING SUBCHANNEL WIDTH          .
C     .            (IFLAG = 1) - PEAK DISCHARGE (cfs) USED FOR         .
C     .                          DETERMINING SUBCHANNEL WIDTH          .
C     .    IDR   - INDEX FOR RILL                                      .
C     .    ISC   - INDEX FOR SUBCHANNEL                                .
C     .    NUMSC - NUMBER OF SUBCHANNELS PER RILL    IDR               .
C     .    NELM  - NUMBER OF ELEMENTS FOR SUBCHANNEL ISC               .
C     .    CLGH  - SUBCHANNEL LENGTH (m)                               .
C     .    CSLP  - SUBCHANNEL SLOPE (m/m)                              .
C     .    FCAR  - CUMMULATIVE FLOW AREA (m**2)                        .
C     .    CWTH  - CHANNEL WIDTH (m)                                   .
C     .    QIN   - BASEFLOW (cfs)                                      .
C     .                                                                .
```

```
C     .    NSLVC - FIRST VALUE IS THE NUMBER OF MASTER CHANNELS FOR    .
C     .            CHANNEL ISC                                         .
C     .            SECOND THROUGH FIFTH ARE THE MASTER CHANNEL NUMBERS .
C     .            ( MAXIMUM OF FOUR )                                 .
C     .                                                                .
C     ..................................................................
C
          READ(IIN2,1001)IDUM1,IDUM2,NELM,CLGH,CSLP,FCAR,CWTH,QIN
          READ(IIN2,1002)(NSLVC(I,ISC),I=1,5)
C
C     ..................................................................
C     .                                                                .
C     .   CONVERT VARIABLES TO ENGLISH UNITS                           .
C     .   1 METER = 3.281 FEET                                         .
C     ..................................................................
C
          EM = 3.281
C
          YIN  = ZERO
          CLGH = CLGH*EM
          CWTH = CWTH*EM
C
C     ..................................................................
C     .                                                                .
C     .    CALCULATE SUBRILL WIDTH                                     .
C     .                                                                .
C     ..................................................................
C
          QMAX=FCAR
          IF(IFLAG.GE.0)QMAX=RAV*FCAR*EM*EM
          IF(CWTH.LE.0)CALL CWIDTH(QMAX,CWTH)
C
C     ..................................................................
C     .                                                                .
C     .    SET CHANNEL PARAMETERS                                      .
C     .                                                                .
C     .    CHR    - THE CHANNEL ROUGHNESS COEFFICIENT                  .
C     .    CHS    - THE CHANNEL SLOPE                                  .
C     .    CHW    - THE CHANNEL BASE WIDTH                             .
C     .    CHZ    - THE CHANNEL SIDE SLOPE                             .
C     .    QFLW   - THE INITIAL CHANNEL FLOW (ASSUMED CONSTANT)        .
C     .                                                                .
C     ..................................................................
C
           CHR(ISC)   = RMANN
           CHS(ISC)   = CSLP
           CHW(ISC)   = CWTH
           CHZ(ISC)   = 0.0
           QFLW(ISC) = QIN
C
C     .................................................................
C     .                                                               .
C     .    CALCULATING THE SUBRILL CONSTANTS.                         .
```

```
C       .                                                              .
C       . NEQ    - NUMBER OF EQUATIONS                                 .
C       . NEL    - NUMBER OF CHANNEL FINITE ELEMENTS                   .
C       . NUMEL  - NUMBER OF CHANNEL FINITE ELEMENTS                   .
C       . NNP    - NUMBER OF CHANNEL NODE POINTS                       .
C       . NUMNP  - NUMBER OF CHANNEL NODE POINTS                       .
C       ................................................................
C
        NEL(ISC) = NELM
        NNP(ISC) = NELM+1
C
        NUMEL = NEL(ISC)
        NUMNP = NNP(ISC)
        NEQ   = NUMNP - 1
C
C       ................................................................
C       .                                                              .
C       .   SET ELEMENT NODE CONNECTIVITY                              .
C       .                                                              .
C       ................................................................
C
        DO 16 IEL = 1,10
          NCON(1,IEL,ISC)=IEL
          NCON(2,IEL,ISC)=IEL+1
16      CONTINUE
C
C       SET HYDROGRAPH FLAGS
C
C       ................................................................
C       .                                                              .
C       . NH     - CHANNEL HYDROGRAPH NUMBER                           .
C       .          = 0; NO UPSTREAM HYDROGRAPH                         .
C       .          = 1; MASTER CHANNEL WITH UPSTREAM HYDROGRAPH        .
C       . NLF    - CHANNEL LATERAL INFLOW NUMBER                       .
C       .          = 0; NO LATERAL INFLOW                              .
C       .          = 1; CHANNEL SUBJECTED TO LATERAL INFLOW            .
C       . NTMC  - NUMBER OF TOTAL MASTER CHANNELS                      .
C       ................................................................
C
        IF (NSLVC(1,ISC) .LE. 0) NH(ISC) = 1
        IF (NSLVC(1,ISC) .GE. 0) NLF(ISC) = 1
        IF (NSLVC(1,ISC) .LE. 0) NTMC = NTMC + 1
C
C       ................................................................
C       .                                                              .
C       .   LOOP OVER SUBRILLS                                         .
C       .                                                              .
C       ................................................................
C
        NEL2 = NUMEL*2
C
        DO 20 I=1,NEL2
C
```

```
C       ...................................................................
C       .                                                                 .
C       .    READ IN PLANE DATA                                           .
C       .                                                                 .
C       .    IEN      - ELEMENT NUMBER                                    .
C       .    IP       - SUBPLANE INDEX (1-RIGHT, 2-LEFT, 0-UPPER          .
C       .    IX       - IP = 1; X COORDINATE OF UPPER ELEMENT BOUNDARY    .
C       .    IX       - IP = 2; X COORDINATE OF LOWER ELEMENT BOUNDARY    .
C       .    IY       - IP = 1; Y COORDINATE OF UPPER ELEMENT BOUNDARY    .
C       .    IY       - IP = 2; Y COORDINATE OF LOWER ELEMENT BOUNDARY    .
C       .    AREA     - SUBELEMENT AREA (m**2)                            .
C       .    PLGN     - SUBELEMENT LENGTH (m)                             .
C       .    PSLP     - SUBELEMENT SLOPE (m/m)                            .
C       .    PRGH     - SUBELEMENT ROUGHNESS (mm)                         .
C       .    FAVL     - CUMMULATIVE FLOW AREA (m**2)                      .
C       .    ELNG     - ELEMENT LENGTH (m)                                .
C       ...................................................................
C
             READ(IIN2,1003)IEN,IP,IX,IY,AREA,PLNG,PSLP,PRGH,FAVL,ELNG
C
C       ...................................................................
C       .                                                                 .
C       .  REVERSE ELEMENT INDEX TO MAKE HYMODRIN COMPATABLE WITH GHIM    .
C       .                                                                 .
C       ...................................................................
C
             IEL = NELM - IEN + 1
C
C       ...................................................................
C       .                                                                 .
C       .    STORE ELEMENT LENGTH                                         .
C       .                                                                 .
C       ...................................................................
C
             ELN(IEL,ISC)    = ELNG
C
C       ...................................................................
C       .                                                                 .
C       .    SET ELEMENT TYPE NELTP(IEL,ISC)= 1 (LINEAR)                  .
C       .                                                                 .
C       ...................................................................
C
             NELTP(IEL,ISC) = 1
C       ...................................................................
C       .                                                                 .
C       .    GENERATE OVERLAND FLOW HYDROGRAPH FOR ELEMENT I              .
C       .                                                                 .
C       ...................................................................
C
12             CALL RUNOFF
C
               DO 15 J=1,NTS
C
```

```
                  IF(IP.EQ.1)QL(IEL,J)   = QO(J)
                  IF(IP.GE.2)QL(IEL,J)   = QL(IEL,J)+QO(J)
C
15            CONTINUE
C
C     ..................................................................
C     .                                                                .
C     .    SET INTIAL LATERAL INFLOW VALUES                            .
C     .                                                                .
C     ..................................................................
C
              IF(IP.GE.2)QLT1(IEL,ISC) = QL(IEL,1)
C
20          CONTINUE
C
C     ..................................................................
C     .                                                                .
C     .    WRITE LATERAL HYDROGRAPHS TO SCRATCH FILES                  .
C     .                                                                .
C     ..................................................................
C
          DO 30 ITS=1,NTS
C            WRITE(IFL(ISC),REC=ITS) (QL(IEL,ITS),IEL=1,NUMEL)
             WRITE(IFL(ISC))         (QL(IEL,ITS),IEL=1,NUMEL)
30        CONTINUE
C
C     ..................................................................
C     .                                                                .
C     .    DETERMINE ELEMENT CHARACTERISTICS                           .
C     .                                                                .
C     ..................................................................
C
        CALL PROFIL
C
        CALL ELEQN
C
        CALL MATRIX
C
C     ..................................................................
C     .                                                                .
C     .    EVALUATE UPSTREAM HYDROGRAPHS                               .
C     .                                                                .
C     ..................................................................
C
        IF (NSLVC(1,ISC) .GT. 0) GO TO 40
C
        READ(IIN2,1003)IEL,IP,IX,IY,AREA,PLNG,PSLP,PRGH,FAVL,ELNG
C
             CALL RUNOFF
C
             DO 35 J=1,NTS
C
                QU(NTMC,J)   = QO(J)
```

```
C
35             CONTINUE
C
40    CONTINUE
C
C     ..................................................................
C     .                                                                .
C     .    OUTPUT CHANNEL CONSTANTS                                    .
C     .                                                                .
C     ..................................................................
C
      CALL COUT
C
C     ..................................................................
C     .                                                                .
C     .    WRITE UPSTREAM HYDROGRAPHS TO SCRATCH FILES                 .
C     .                                                                .
C     ..................................................................
C
        DO 45 ITS=1,NTS
           WRITE(IF1)    (QU(ITMC,ITS),ITMC=1,NTMC)
   45   CONTINUE
C
C     ..................................................................
C     .                                                                .
C     .    ROUTE RILL FLOWS                                            .
C     .                                                                .
C     ..................................................................
C
      REWIND IF1
C
      DO 60 ISC=1,3
         IF3 = IFL(ISC)
         REWIND IF3
60    CONTINUE
C
      CALL CROUTE
C
50    CONTINUE
C
C     ..................................................................
C     .                                                                .
C     .    GENERATE NULLRILLSHED HYDROGRAPH                            .
C     .                                                                .
C     ..................................................................
C
      READ(IIN2,1004)IDUM,NUMSC
      READ(IIN2,1001)IDUM1,IDUM2,NELM,CLGH,CSLP,FCAR,CWTH,QIN
      READ(IIN2,1002)(NSLVC(I,ISC),I=1,5)
C
      READ(IIN2,1003)IEL,IP,IX,IY,AREA,PLNG,PSLP,PRGH,FAVL,ELNG
C
      CALL RUNOFF
```

```
C
      END
      BLOCK DATA
C.................................................................
C.                                                               .
C.        CHANNEL NETWORK FLOW MATRICES INITIALIZATION ROUTINE   .
C.                                                               .
C.................................................................
C.                                                               .
C.    THE PURPOSE OF THIS BLOCK DATA SUBPROGRAM IS TO INITIALIZE  .
C.    THE CHANNEL NETWORK FLOW DATA MATRICES.                    .
C.                                                               .
C.................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
      COMMON/FLWMAT/AFM(18),QFV(6),VFM(18)
C
C     INITIALIZING THE LINEAR, QUADRATIC AND CUBIC LAGRANGIAN
C     ELEMENT AREA FLOW AND VOLUME FLOW MATRICES; AND THE
C     THE LATERAL INFLOW VECTOR.
C
      DATA AFM/ 0.333333333333333, 0.16666666666667,0.D0,
     1          0.166666666666667, 0.333333333333333,4*0.D0,
     2          0.133333333333333, 0.066666666666667,-0.033333333333333,
     2          0.066666666666667, 0.500000000000000, 0.066666666666667,
     2         -0.033333333333333, 0.066666666666667, 0.133333333333333/
C
      DATA VFM/-0.50D0,-0.50D0,0.D0,0.50D0,0.50D0,4*0.D0,
     2         -0.500000000000000,-0.666666666666667, 0.160066666666667,
     2          0.666666666666667, 0.000000000000000,-0.666666666666667,
     2         -0.166666666666667, 0.666666666666667, 0.500000000000000/
C
      DATA QFV/ 0.5D0,0.5D0,0.D0,
     2          0.166666666666667, 0.666666666666667, 0.166666666666667/
C
      END
      SUBROUTINE SDATA
C
C     .................................................................
C     .                                                               .
C     .    THIS SUBROUTINE READS IN ALL CONTROL DATA AND INITIALIZES  .
C     .    THE RAINFALL EXCESS VECTORS                                .
C     .                                                               .
C     .................................................................
C
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
C
      COMMON/CDATA/CLGH,CSLP,CWTH,FCAR,RMANN,TRACF,YIN
      COMMON/RDATA/ONE,R23,R35,R53,ZERO
      COMMON/FILES/IIN1,INN2,IOUT
      COMMON/FDATA/SAV1,SAV2,SKFS1,SKFS2,SLI,SMCI,POR1,POR2
      COMMON/TIME/DT,RTH,RTH1,RTT,DELTA,IFDC,ITS,NFDC,NRAIN,NTS
```

```
      COMMON/PDATA/AREA,PLNG,PRGH,PSLP,CLAY,SAND
      COMMON/RAIND/AKE(2000),HY(60),RE(2000),RN(2000),RAV
      COMMON/ITDAT/ANORM,BCT,DWT,MBCIT,MDWIT,NBCS,NCSC
C
      CHARACTER*72 ITLE(2)
C
 1001 FORMAT(A72)
 1002 FORMAT(16I5)
 1003 FORMAT(8F10.1)
 1004 FORMAT(I10,F10.1)
 1005 FORMAT(6F10.2)
 1006 FORMAT(6F10.2)
 1007 FORMAT(4F10.2)
C
2001   FORMAT(//4X,1H+,74(1H-),1H+/4X,2H+ ,A72,2H +/4X,2H+ ,A72,2H +
     1/4X,1H+,74(1H-),1H+)
2002   FORMAT(4X,1H+,74X,1H+
     1/4X,1H+,5X,28HTHE NUMBER OF RILL NETWORKS ,36(1H.),I4,5X,1H+
     2/4X,1H+,5X,25HTHE NUMBER OF TIME STEPS ,35(1H.),I4,5X,1H+
     3/4X,1H+,5X,33HTHE FLOW DEPTH PROFILE TIME STEP ,27(1H.),I4,5X,1H+
     4/4X,1H+,5X,40HTHE NUMBER OF DIFFUSION WAVE ITERATIONS ,20(1H.),
     4 I4,5X,1H+
     5/4X,1H+,5X,44HTHE NUMBER OF BOUNDARY CONDITION ITERATIONS ,
     5 16(1H.),I4,5X,1H+/4X,1H+,74X,1H+)
2003  FORMAT(4X,1H+,5X,19HTHE TIME INCREMENT ,35(1H.),1PD10.3,5X,1H+
     1/4X,1H+,5X,31HTHE TIME WEIGHTING COEFFICIENT ,23(1H.),
     1 1PD10.3,5X,1H+
     2/4X,1H+,5X,41HTHE DIFFUSION WAVE CONVERGENCE TOLERANCE ,13(1H.),
     2 1PD10.3,5X,1H+
     3/4X,1H+,5X,45HTHE BOUNDARY CONDITION CONVERGENCE TOLERANCE ,
     3 9(1H.),1PD10.3,5X,1H+/4X,1H+,74X,1H+/4X,1H+,74(1H-),1H+)
C
C      ...................................................................
C      .                                                                 .
C      .     READ TITLE INFORMATION                                      .
C      .                                                                 .
C      ...................................................................
C
      READ(IIN1,1001) ITLE
C
C....................................................................
C.                                                                  .
C.          READ PLOT DATA                                          .
C.                                                                  .
C.          NR  - NUMBER OF RILLS                                   .
C.          NTS - NUMBER OF TIME STEPS                              .
C.          IR  - OPTION FOR RILL WIDTHS                            .
C.                0 - FCAR - CUMMLUATIVE FLOW AREA                  .
C.                1 - FCAR - PEAK DISCHARGE                         .
C.                                                                  .
C.          DT    - TIME STEP IN SECONDS                            .
C.          TRACF - CRITICAL TRACTIVE FORCE OF SOIL                 .
C.          RMANN - MANNINGS ROUGHNESS COEFFICIENT                  .
```

```
C.                                                                    .
C.      NFDC  - TIME STEP FLOW DEPTH PROFILE OUTPUT NUMBER            .
C.              = 0 ==> OUTFLOW HYDROGRAPH ONLY                       .
C.              = I ==> FLOW DEPTH PROFILE FOR EVERY I TIME STEP      .
C.                     OUTPUTTED                                      .
C.      MDWIT - THE MAXIMUM NUMBER OF DIFFUSION WAVE ITERATIONS       .
C.                (GENERALLY INPUT  10 < MDWIT < 50; DEFAULT = 20)    .
C.      MBCIT - THE MAXIMNUM NUMBER OF DOWNSTREAM BOUNDARY CONDITION  .
C.                ITERATIONS                                          .
C.                (GENERALY INPUT 4 < MBCIT < 10; DEFAULT = 6)        .
C.                                                                    .
C.      RTH   - THE TIME WEIGHTING COEFFICIENT BETWEEN 0.50 AND 1.00  .
C.                ( 0.50 <= RTH <= 1.00; DEFAULT 0.50D0 )             .
C.      DWT   - RELATIVE DIFFUSION WAVE TOLERANCE                     .
C.                ( GENERALLY INPUT  1.D-8 < TOL < 1.D-6 )            .
C.                ( DEFAULT = 1.D-8 )                                 .
C.      BCT   - RELATIVE BOUNDARY CONDITION TOLERANCE                 .
C.                ( GENERALLY INPUT  1.D-8 < TOL < 1.D-6 )            .
C.                ( DEFAULT = 1.D-8 )                                 .
C.                                                                    .
C......................................................................
C
      READ(IIN1,1002) NR,IR,NTS,NFDC,MDWIT,MBCIT
      READ(IIN1,1005) RTH,DWT,BCT,DT,TRACF,RMANN
C
C     CHECK INPUT DATA AND GENERATE TIME INTERPOLATION CONSTANTS.
C
      IF (MDWIT .LT. 10   .OR.  MDWIT .GT. 50) MDWIT = 20
      IF (MBCIT .LT.  4   .OR.  MBCIT .GT. 10) MBCIT = 6
      IF (RTH .LT. 0.5D0   .OR.  RTH .GT. ONE) RTH = 0.5D0
      IF (DWT .LT. 1.D-10   .OR.  DWT .GT. 1.D-6) DWT = 1.D-8
      IF (BCT .LT. 1.D-10   .OR.  BCT .GT. 1.D-6) BCT = 1.D-8
c
      RTH1 = ONE - RTH
      RTT  = RTH*DT
C
      WRITE(IOUT,2001) ITLE
      WRITE(IOUT,2002) NR,NTS,NFDC,MDWIT,MBCIT
      WRITE(IOUT,2003) DT,RTH,DWT,BCT
C
C     ..................................................................
C     .                                                                .
C     .    READ RAINFALL DATA                                          .
C     .                                                                .
C     ..................................................................
C
      READ(IIN1,1004)NRAIN,DELTA
      READ(IIN1,1003)(HY(I),I=1,NRAIN)
C
C     ..................................................................
C     .                                                                .
C     .    DISAGGREGATE RAINFALL DATA                                  .
C     .                                                                .
```

```
C     ..................................................................
C
      CALL DRAIN
C
C     ..................................................................
C     .                                                                .
C     .    GENERATE CUMMULATIVE KINETIC ENERGY ARRAY                   .
C     .                                                                .
C     ..................................................................
C
      CALL KINETIC
C
C     ..................................................................
C     .                                                                .
C     .    READ IN RAINFALL EXCESS DATA                                .
C     .                                                                .
C     ..................................................................
C
C
C     SAV1  - AVERAGE CAPILLARY SUCTION PRESSURES FOR UPPER LAYER (cm)
C     SAV2  - AVERAGE CAPILLARY SUCTION PRESSURES FOR UPPER LAYER (cm)
C     SKFS1 - THE HYDRAULIC CONDUCTIVITIES FOR THE UPPER LAYER (cm/hr)
C     SKFS2 - THE HYDRAULIC CONDUCTIVITIES FOR THE UPPER LAYER (cm/hr)
C     SLI   - THE DEPTH OF THE UPPER LAYER (cm)
C     POR1  - EFFECTIVE POROSITIES FOR UPPER LAYER (cm^3/cm^3),
C     POR2  - EFFECTIVE POROSITIES FOR LOWERR LAYER (cm^3/cm^3),
C     SMCI  - THE INITIAL SOIL MOISTURE CONTENT (%).
C     CLAY  - PERCENT CLAY OF SOIL
C     SAND  - PERCENT SAND OF SOIL
C
      READ(IIN1,1007) SLI,SMCI,CLAY,SAND
      READ(IIN1,1006) SAV1,SAV2,SKFS1,SKFS2,POR1,POR2
C
      RETURN
      END
      SUBROUTINE DRAIN
C
C.....................................................................
C.                                                                   .
C.                      RAINFALL DISSAGGREGATION                     .
C.                                                                   .
C.....................................................................
C.                                                                   .
C.    THE PURPOSE OF THIS SUBROUTINE IS TO DISSAGGREGATE THE INPUT   .
C.    RAINFALL SERIES INTO A SERIES ASSOCIATED WITH THE SELECTED     .
C.    COMPUTATION TIME STEP                                          .
C.....................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
C
      COMMON/RAIND/AKE(2000),HY(60),RE(2000),RN(2000),RAV
      COMMON/TIME/DT,RTH,RTH1,RTT,DELTA,IFDC,ITS,NFDC,NRAIN,NTS
C
```

```
C     RAINFALL DATA IN cm PER DELTA (sec)
C
C     DELTA (sec) = INPUT RAINFALL TIME INCREMENT
C     DT (sec)    = COMPUTATIONAL  TIME INCREMENT
C
C     CALCULATE NUMBER OF INCREMENTS TO SUBDIVIDE RAINFALL VOLUME
C     ASSOCIATED WITH EACH INPUT INTERVAL (DELTA)
C
            INC = DELTA/DT
C
C           CALCULATE THE HYETOGRAPH AT TIME T + DT.
C
            ICT = 0
            DO 20 I=1,NRAIN
               DO 10 J=1,INC
                  ICT = ICT + 1
                  RN(ICT) = HY(I)*DT/DELTA
10             CONTINUE
20          CONTINUE
C
C           CHECK ON TIME LIMIT OF INPUT HYETOGRAPHS.
C
            IF (ICT .GE. NTS) GO TO 40
C
C           SET REMAINDER OF THE HYETOGRAPH TO ZERO.
C
            DO 30 ITS=ICT,NTS
               RN(ITS) = ZERO
30          CONTINUE
40    CONTINUE
C
C           DETERMINE AVERAGE RAINFALL INTENSITY (RAV)
C
            DO 50 ITS = 1,ICT
            SR = SR + RN(ITS)
50          CONTINUE
            RAV = SR/ICT
      RETURN
      END
      SUBROUTINE KINETIC
C.....................................................................
C.                                                                   .
C.                      KINETIC ENERGY  ROUTINE                      .
C.                                                                   .
C.....................................................................
C.                                                                   .
C.    THE PURPOSE OF THIS SUBROUTINE IS TO CACULATE THE CUMMULATIVE  .
C.    KINETIC ENERGY ASSOCIATED WITH A PARTICULAR RAINFALL SERIES    .
C.....................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
C
      COMMON/RAIND/AKE(2000),HY(60),RE(2000),RN(2000),RAV
```

```
      COMMON/RDATA/ONE,R23,R35,R53,ZERO
      COMMON/TIME/DT,RTH,RTH1,RTT,DELTA,IFDC,ITS,NFDC,NRAIN,NTS
C
      SUMEK = ZERO
C
      DO 10 I=1,NTS
C
C     DETERMINE RAINFALL INTENSITY mm/hr.
C
      RAINC=3600.0*RN(I)*2.54/DT
C
C     CALCULATE EK, THE RAINFALL KINETIC ENERGY APPLIED DURING
C     ONE TIME STEP (joules/sqcm per cm of rain).
C
      EK = .0001*(11.9 + (8.73*DLOG(RAINC)))*RN(I)
C
      SUMEK = SUMEK + EK
C
      AKE(I)=SUMEK
C
10    CONTINUE
      RETURN
      END
      SUBROUTINE RUNOFF
C.....................................................................
C.                                                                   .
C.                     RUNOFF       ROUTINE                          .
C.                                                                   .
C.....................................................................
C.                                                                   .
C.    THE PURPOSE  OF THIS SUBROUTINE  IS TO  CALCULATE  THE         .
C.    RUNOFF EXCESS AND WATERSHED HYDROGRAPH USING THE GREEN         .
C.    AMPT INFILTRATION  MODEL AND  THE NONLINEAR  RESERVOIR         .
C.    RUNOFF MODEL                                                   .
C.                                                                   .
C.....................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)

      COMMON/FLOWA/QO(1000)
      COMMON/CDATA/CLGH,CSLP,CWTH,FCAR,RMANN,TRACF,YIN
      COMMON/PDATA/AREA,PLNG,PRGH,PSLP,CLAY,SAND
      COMMON/FDATA/SAV1,SAV2,SKFS1,SKFS2,SLI,SMCI,POR1,POR2
      COMMON/RDATA/ONE,R23,R35,R53,ZERO
      COMMON/RAIND/AKE(2000),HY(60),RE(2000),RN(2000),RAV
      COMMON/TIME/DT,RTH,RTH1,RTT,DELTA,IFDC,ITS,NFDC,NRAIN,NTS
      COMMON/GRAMP/F(2000)
C
C     **********************************
C     * DEFINE INFILTRATION VARIABLES *
C     **********************************
C
C     SLI   = THE DEPTH OF THE UPPER LAYER (cm)
```

```
C      SMCI   = THE INITIAL SOIL MOISTURE CONTENT.
C      SAV1   = AVERAGE CAPILLARY SUCTION PRESSURES FOR UPPER LAYER (cm)
C      SAV2   = AVERAGE CAPILLARY SUCTION PRESSURES FOR UPPER LAYER (cm)
C      SKFS1 = THE HYDRAULIC CONDUCTIVITIES FOR THE UPPER LAYER (cm/hr)
C      SKFS2 = THE HYDRAULIC CONDUCTIVITIES FOR THE UPPER LAYER (cm/hr)
C      POR1   = EFFECTIVE POROSITIES FOR UPPER LAYER (cm^3/cm^3),
C      POR2   = EFFECTIVE POROSITIES FOR LOWER LAYER (cm^3/cm^3),
C
       FH1 = SLI
       FC1 = SAV1
       FC2 = SAV2
       FK1 = SKFS1
       FK2 = SKFS2
       FS1 = POR1-SMCI
       FS2 = POR2-SMCI
       SD1 = FC1*FS1
       SD2 = FC2*FS2
C
       IF1    =   ZERO
       FZ     =   ZERO
       DF     =   ZERO
       F1     = 0.0001
       F2     = 0.0001
       HR     = DT/3600
       FMAX   = FH1*FS1
       DFMIN = FK1*HR
       CMFT   = 2.54*12.0
C
       SUMR = ZERO
       SUME = ZERO
       SUMF = ZERO
       SUMQ = ZERO
       DSTR = ZERO
C
C      ***************************
C      * DEFINE RUNOFF VARIABLES *
C      ***************************
C
       D1    = ZERO
       DD    = ZERO
       DS    = ZERO
       CZ    = ZERO
C
C      SET INTIAL DEPRESSION STORAGE
C
       DS  = DSTR
       DSM = DS/CMFT
C
C      CONVERT INITIAL RANDOM ROUGHNESS FROM mm TO cm.
C
       RRO=PRGH/10
C
C      SELECT ROUGHNESS EQUATION
```

```
C
      IF(SAND.GT.ZERO.AND.CLAY.GT.ZERO)CALL CHEZY (CZ)
C
      IF(SAND.LE.ZERO)PMANN = RMANN
C
C     CHEZY'S EQUATION
C
      COE1 = .750
      ZXP1 = 1.50
      ZXP2 = 0.50
      WCON = -(AREA/PLNG)*CZ*SQRT(PSLP)/AREA
C
C     MANNING'S EQUATION
C
      IF(PMANN.GT.ZERO)COE1 = 5.0/6.0
      IF(PMANN.GT.ZERO)ZXP1 = 5.0/3.0
      IF(PMANN.GT.ZERO)ZXP2 = 2.0/3.0
      IF(PMANN.GT.ZERO)WCON = -(AREA/PLNG)*1.49*SQRT(PSLP)/(PMANN*AREA)
C
C     *****************************
C     * BEGIN COMPUTATION DO LOOP *
C     *****************************
C
      DO 100 I=1,NTS
C
C     ****************************************
C     * CALCULATE CURRENT DEPRESSION STORAGE *
C     ****************************************
C
      CKE=AKE(I)
C
      IF(RRO.GT.ZERO)CALL STORE (CKE,RRO,RRT,DS)
C
      IF(RRO.GT.ZERO)DSM=DS/CMFT
C
C     *****************************************************************
C     *     DETERMINE INFILTRATION USING THE GREEN AMPT EQUATION      *
C     *****************************************************************
C
C
C     CHECK FOR IMPERVIOUS SURFACE
C
      IF(FH1.GT.ZERO)GO TO 10
      DF    = ZERO
      RE(I) = RN(I)
      GO TO 60
C
C     DETERMINE INCREMENTAL INFILTRATION
C
10    CONTINUE
C
      PP    = RN(I) + D1
      ALP1 = (SD1+(D1*FS1))
```

```
C
      A    = (((2*F1)-(FK1*HR))*0.5)**2
      B    = 2*FK1*HR
      DF   = (FK1*HR*.5)-F1+(SQRT(A+(B*(ALP1+F1))))
      DF   = (FK1*HR)+(ALP1*LOG(1+(DF/(ALP1+F1))))
C
      IF(DF.LE.DFMIN)DF=DFMIN
C
      IF(DF.GT.PP)DF = PP
C
      F1 = F1 + DF
      IF(F1.GE.FMAX)F1=FMAX
C
C     IF(F1.GE.FMAX)   GO TO 30
C
      DDF = PP-DF
C
      IF(DDF.GT.DSM)GO TO 20
C
C     CASE I: D1 + RN(I) - DF < DS
C
      D1 = D1 + RN(I) - DF
      IF(D1.LE.ZERO)D1 = ZERO
      RE(I) = ZERO
      QO(I) = ZERO
      QQ    = ZERO
      GO TO 50
C
C     CASE II: D1 + RN(I) - DF > DS
C
20    RE(I)=RN(I)-DF
C
C     *******************************************************
C     * DETERMINE RUNOFF USING NONLINEAR RESERVIOR ALGORITHM *
C     *******************************************************
C
      QO(I)=ZERO
C
C     CONVERT RAINFALL EXCESS FROM (cm) TO (ft)
C
      DD = RE(I)/CMFT
C
C     EXPRESS RAINFALL EXCESS IN (ft/sec)
C
      E = DD/DT
C
      IC=0
      DD=E
C
30    IC=IC+1
C
      IF(IC.GE.10)GO TO 40
C
```

```
      D = (D1/CMFT) - (DS/CMFT)+(DD/2.0)
      IF(D.LE.ZERO)D=ZERO
C
      AAA = DT*((WCON*(D**ZXP1))+E)
      FZ = DD - AAA
      BBB = DT*(COE1*WCON*(D**ZXP2))
      ZF = 1 - BBB
C
      DDD=DD-(FZ/ZF)
      DIF=ABS(DDD-DD)
C
      DD=DDD
C
      IF(DIF.GE.0.001)GO TO 30
C
40    QO(I)=AREA*(E-(DD/DT))
      QQ=(E-(DD/DT))*DT*CMFT
      IF(QO(I).LE.ZERO)QO(I)=ZERO
      D1=D1+(DD*CMFT)
      IF(D1.LE.ZERO)D1=ZERO
C
C     OUTPUT RUNOFF RESULTS
C
50    CONTINUE
C
      RETURN
      END
      SUBROUTINE STORE(CKE,RR0,RRT,DS)
C
C..................................................................
C.                                                               .
C.                     SURFACE STORAGE ROUTINE                   .
C.                                                               .
C..................................................................
C.                                                               .
C.    THE PURPOSE OF THIS SUBROUTINE IS TO CACULATE THE SURFACE  .
C.    STORAGE TIME VECTORS FOR THE SUBPLANE.                     .
C..................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
C
      COMMON/PDATA/AREA,PLNG,PRGH,PSLP,CLAY,SAND
C
      DIMENSION RRE(6)
C
      DATA RRE(1),RRE(2),RRE(3),RRE(4),RRE(5),RRE(6)/
     X0.,.8,1.6,2.4,3.2,4./
C
C     LOOP TO CALCULATE AVAILABLE SURFACE STORAGE VOLUME IN TIME.
C
C
C        CALCULATE CURRENT RANDOM ROUGHNESS IN (cm)
C              (Normal values range from .8 to .4)
```

```
C
C          CKE = CUMMULATIVE KINETIC ENERGY (joules/sqcm)
C                (Normal values range from 0 to .50)
C
      RRT=RR0*0.9644*EXP(-1.989*CKE)
C
      IF(RRT.GT.4.0)RRT=4.0
C
      S=PSLP*100.
C
      IF(S.GT.20)S=20
C
      IF(RRT.GT.0.8)GO TO 10
C
      R1=ZERO
      R2=0.8
      S1=ZERO
      C21 = 1.042E-03
      C22 = 1.208e-02
      C23 = 5.E-05
      C24 = 0.2000
      S2=C21*S**3. - C22*S*S + C23*S + C24
      IF(S2.LE.ZERO)S2=ZERO
      GO TO 50
C
10    IF(RRT.GT.1.6)GO TO 20
C
      R1=0.8
      R2=1.6
      C11 = 1.042E-03
      C12 = 1.208e-02
      C13 = 5.E-05
      C14 = 0.2000
      S1=C11*S**3. - C12*S*S + C13*S + C14
      IF(S1.LE.ZERO)S1=ZERO
      C21 = 8.396E-04
      C22 = 1.319E-02
      C23 = 8.018E-03
      C24 = 0.41
      S2=C21*S**3. - C22*S*S + C23*S + C24
      IF(S2.LE.ZERO)S2=ZERO
      GO TO 50
C
20    IF(RRT.GT.2.4)GO TO 30
C
      R1=1.6
      R2=2.4
      C11 = 8.396E-04
      C12 = 1.319E-02
      C13 = 8.018E-03
      C14 = 0.41
      S1=C11*S**3. - C12*S*S + C13*S + C14
      IF(S1.LE.ZERO)S1=ZERO
```

```
      C21 = 6.008E-04
      C22 = 1.146E-02
      C23 = 5.163E-04
      C24 = 0.62
      S2=C21*S**3. - C22*S*S + C23*S + C24
      IF(S2.LE.ZERO)S2=ZERO
      GO TO 50
C
30    IF(RRT.GT.3.2)GO TO 40
C
      R1=2.4
      R2=3.2
      C11 = 6.008E-04
      C12 = 1.146E-02
      C13 = 5.163E-04
      C14 = 0.62
      S1=C11*S**3. - C12*S*S + C13*S + C14
      IF(S1.LE.ZERO)S1=ZERO
      C21 = 5.167E-04
      C22 = 1.165E-02
      C23 = 3.833E-03
      C24 = 0.84
      S2=C21*S**3. - C22*S*S + C23*S + C24
      IF(S2.LE.ZERO)S2=ZERO
      GO TO 50
C
40    CONTINUE
C
      R1=3.2
      R2=4.0
      C11 = 5.167E-04
      C12 = 1.165E-02
      C13 = 3.833E-03
      C14 = 0.84
      S1=C11*S**3. - C12*S*S + C13*S + C14
      IF(S1.LE.ZERO)S1=ZERO
      C21 = 3.056E-04
      C22 = 8.528E-03
      C23 = -3.278E-03
      C24 = 1.06
      S2=C21*S**3. - C22*S*S + C23*S + C24
      IF(S2.LE.ZERO)S2=ZERO
C
50    DS=((RRT-R1)/(R2-R1)*(S2-S1))+S1
C
      IF(DS.LE.0.0)DS=0.0
C
      RETURN
      END
      SUBROUTINE CHEZY(CZ)
C
C....................................................................
C.                                                                  .
```

```
C.                        CHEZY ROUGHNESS COEFFICIENT                        .
C.                                                                           .
C............................................................................
C.                                                                           .
C.     THE PURPOSE OF THIS SUBROUTINE IS TO CACULATE THE CHEZY               .
C.     ROUGHNESS COEFFICIENT                                                 .
C.                                                                           .
C.     SOURCE: HYDRAULICS OF OVERALND FLOW - GILLEY, GINKNER, NEARING         .
C.     AND LANE (WEPP MODEL DOCUMENTATION 1989)                              .
C............................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
C
      COMMON/PDATA/AREA,PLNG,PRGH,PSLP,CLAY,SAND
C
      FO=4.0*(3.42**CLAY)/(12.42**SAND)
C
      CZ=(8*32.2/FO)**.5
C
      RETURN
      END
      SUBROUTINE CWIDTH(QMAX,WEQ)
C
C............................................................................
C.                                                                           .
C.                 CHANNEL WIDTH ROUTINE                                     .
C.                                                                           .
C............................................................................
C.                                                                           .
C.     THE PURPOSE OF THIS SUBROUTINE IS TO CALCULATE THE CHEZY'S C          .
C.     FRICTION FACTOR FOR OVERLAND FLOW AS IT VARIES WITH TIME.             .
C.                                                                           .
C............................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
C
      COMMON/CDATA/CLGH,CSLP,CWTH,FCAR,RMANN,TRACF,YIN
C
      GAMMA=9802.0
C
C     DETERMINE CONVEYANCE FUNCTION G(x*)
C
      AA=((QMAX*RMANN/(CSLP**.5))**(3.0/8.0))
      BB=GAMMA*CSLP/TRACF
      GXSTAR=AA*BB
C
      IF(GXSTAR.GT.40.0)GXSTAR = 40.0
      IF(GXSTAR.LT.1.80)GXSTAR = 1.80
C
      GG = .1692282
      XSTAR = .3696101*(GXSTAR**(-1.115031))
      IF(GXSTAR.LT.2.0)XSTAR = ((2.0 - GXSTAR)*(.30 - GG)/.20)+GG
C
```

```
      WSTAR = .74 - 1.475*XSTAR
      RSTAR = .16 - .07833343*XSTAR + .7250027*(XSTAR**2) - 5.125016*(X
     1STAR**3) + 7.5*(XSTAR**4) - 4.166693*(XSTAR**5)
C
      WEQ=(QMAX*RMANN/SQRT(CSLP))**(3.0/8.0)*(WSTAR/(RSTAR)**(5.0/8.0))
C
      RETURN
      END
      SUBROUTINE COUT
C.....................................................................
C.                                                                   .
C.                   SUBRILL OUTPUT SUMMARY                          .
C.                                                                   .
C.....................................................................
C.                                                                   .
C.      CHR   - THE CHANNEL ROUGHNESS COEFFICIENT                    .
C.      CHS   - THE CHANNEL SLOPE                                    .
C.      CHW   - THE CHANNEL BASE WIDTH                               .
C.      CHZ   - THE CHANNEL SIDE SLOPE                               .
C.      QFLW  - THE INITIAL CHANNEL FLOW (ASSUMED CONSTANT)          .
C.                                                                   .
C.....................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
C
      COMMON/FILES/IIN1,INN2,IOUT
      COMMON/ICHDAT/IDWIT(12),NLF(12),NEL(12),NH(12),NNP(12),NSLVC(5,12)
      COMMON/IDATA/IEL,ISC,NEND,NEQ,NTMC,NUMEL,NUMNP,NUMSC
      COMMON/RDATA/ONE,R23,R35,R53,ZERO
      COMMON/RLOG/AFAC,BACK
      COMMON/RCHDAT/CHR(12),CHS(12),CHW(12),CHZ(12),DSQ(12),WS(12)
C
      LOGICAL AFAC,BACK
C
1001  FORMAT(16I5)
1002  FORMAT(8F10.0)
2001  FORMAT(//21X,'  ***** S U B C H A N N E L   D A T A  *****',
     1//10X,' CHANNEL  NUMBER  OF   NUMBER OF    MASTER CHANNEL',
     1 4X,'HY      LF'
     2/10X,'  NUMBER   ELEMENTS       NODES          NUMBERS',
     2 6X,' CODE   CODE',/)
2002  FORMAT(10X,I4,7X,I4,9X,I4,8X,4I3,4X,I4,3X,I4)
2003  FORMAT(//15X,
     1 ' CHANNEL   CHANNEL     CHANNEL    CHANNEL     CHANNEL',/15X,
     2 ' NUMBER   ROUGHNESS     SLOPE      WIDTH    SIDE  SLOPE',/)
2004  FORMAT(15X,I4,5X,1PD9.3,3X,1PD9.3,2X,1PD9.3,3X,1PD9.3)
C
C     OUTPUTTING THE CHANNEL DATA AND CALCULATING THE CHANNEL
C     CONSTANTS.
C
      WRITE(IOUT,2001)
C
      DO 20 KISC=1,NUMSC
```

```
         WRITE(IOUT,2002) KISC,NEL(KISC),NNP(KISC),(NSLVC(I,KISC),I=2,5),
     1                    NH(KISC),NLF(KISC)
20    CONTINUE
C
      WRITE(IOUT,2003)
C
      DO 30 KISC=1,NUMSC
         WRITE(IOUT,2004) KISC,CHR(KISC),CHS(KISC),CHW(KISC),CHZ(KISC)
         CHR(KISC) = CHR(KISC)/1.486D0
         DSQ(KISC) = DSQRT(ONE + CHZ(KISC)*CHZ(KISC))
         WS(KISC) = CHW(KISC)*CHW(KISC)
30    CONTINUE
C
      RETURN
      END
      SUBROUTINE PROFIL
C.....................................................................
C.                                                                   .
C.                          PROFILE ROUTINE                          .
C.                                                                   .
C.....................................................................
C.                                                                   .
C.    THE PURPOSE OF THIS SUBROUTINE IS TO CALCULATE THE NUMBER OF   .
C.    EQUATIONS AND ESTABLISH THE DIAGONAL ENTRY ADDRESSES.          .
C.                                                                   .
C.....................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
      COMMON/IDATA/IEL,ISC,NEND,NEQ,NTMC,NUMEL,NUMNP,NUMSC
      COMMON/EQN/A(25,12),RHS(11,12),JD(11,12)
      COMMON/IELMNT/LD(3,10,12),NCON(3,10,12),NELTP(10,12)
      COMMON/INODE/ID(11)
C
C     CALCULATING THE NUMBER OF EQUATIONS AND INITIALIZING THE NODAL
C     EQUATION NUMBER ARRAY.
C
      DO 10 I=1,NEQ
         ID(I) = I
10    CONTINUE
      ID(NUMNP) = 0
C
C     CALCULATING THE COLUMN HEIGHTS.
C
      DO 40 IEL=1,NUMEL
         NEND = NELTP(IEL,ISC) + 1
         DO 30 I=1,NEND
            II = ID(NCON(I,IEL,ISC))
            IF (II .GT. 0) THEN
               DO 20 J=I,NEND
                  JJ = ID(NCON(J,IEL,ISC))
                  IF (JJ .GT. 0) THEN
                     M = MAX0(II,JJ)
                     JD(M,ISC) = MAX0(JD(M,ISC),IABS(II-JJ))
```

```
                  ENDIF
20             CONTINUE
            ENDIF
30       CONTINUE
40    CONTINUE
C
C     CALCULATING THE DIAGONAL ADDRESSES WITHIN THE PROFILE.
C
      JD(1,ISC) = 1
      IF (NEQ .EQ. 1) RETURN
      DO 50 N=2,NEQ
         JD(N,ISC) = JD(N,ISC) + JD(N-1,ISC) + 1
50    CONTINUE
C
      RETURN
      END
      SUBROUTINE ELEQN
C.....................................................................
C.                                                                   .
C.                  ELEMENT EQUATION NUMBER ROUTINE                  .
C.                                                                   .
C.....................................................................
C.                                                                   .
C.    THE PURPOSE OF THIS SUBROUTINE IS TO GENERATE THE ELEMENT      .
C.    EQUATION NUMBER ARRAY.                                         .
C.                                                                   .
C.....................................................................
C
      IMPLICIT INTEGER*4(I-N)
      COMMON/IDATA/IEL,ISC,NEND,NEQ,NTMC,NUMEL,NUMNP,NUMSC
      COMMON/IELMNT/LD(3,10,12),NCON(3,10,12),NELTP(10,12)
      COMMON/INODE/ID(11)
C
C     LOOPING OVER THE ELEMENTS.
C
      DO 20 IEL=1,NUMEL
         NEND = NELTP(IEL,ISC) + 1
C
C        DETERMINING THE ELEMENT EQUATION NUMBER ARRAY.
C
         DO 10 NOD=1,NEND
            LD(NOD,IEL,ISC) = ID(NCON(NOD,IEL,ISC))
10       CONTINUE
C
C        WRITE(6,9001)(LD(I,IEL,ISC),I=1,NEND)
9001     FORMAT(3I5)
20    CONTINUE
C
      RETURN
      END
      SUBROUTINE MATRIX
C.....................................................................
C.                                                                   .
```

```
C.                    FLOW AREA MATRIX GENERATION ROUTINE                  .
C.                                                                         .
C...........................................................................
C.                                                                         .
C.    THE PURPOSE OF THIS SUBROUTINE IS TO GENERATE THE CHANNEL            .
C.    SYSTEM FLOW AREA MATRICES.                                           .
C.                                                                         .
C...........................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
      COMMON/IDATA/IEL,ISC,NEND,NEQ,NTMC,NUMEL,NUMNP,NUMSC
      COMMON/RDATA/ONE,R23,R35,R53,ZERO
      COMMON/FLWMAT/AFM(3,3,2),QFV(3,2),VFM(3,3,2)
      COMMON/IELMNT/LD(3,10,12),NCON(3,10,12),NELTP(10,12)
      COMMON/EQN/A(25,12),RHS(11,12),JD(11,12)
      COMMON/RELMNT/ELN(10,12),QLT1(10,12),QLT2(10,12)
      DIMENSION EFM(3,3)
C
C     INITIALIZING THE SYSTEM COEFFICIENT MATRIX.
C
      NAD = JD(NEQ,ISC)
      DO 10 I=1,NAD
         A(I,ISC) = ZERO
10    CONTINUE
C
C     LOOPING OVER THE ELEMENTS.
C
      DO 30 IEL=1,NUMEL
         IELTP = NELTP(IEL,ISC)
         NEND = IELTP + 1
C
C        CALCULATING THE ELEMENT FLOW AREA MATRIX.
C
         DO 20 J=1,NEND
         DO 20 I=1,NEND
            EFM(I,J) = ELN(IEL,ISC)*AFM(I,J,IELTP)
20       CONTINUE
C
C        ASSEMBLING THE ELEMENT FLOW AREA MATRICES INTO THE SYSTEM
C        COEFFICIENT MATRIX.
C
         CALL ADDSTF(EFM)
30    CONTINUE
C
C     FACTORIZING THE CHANNEL FLOW AREA MATRIX.
C
      CALL ACTCOL(A(1,ISC),JD(1,ISC))
C
      RETURN
      END
      SUBROUTINE ADDSTF(EFM)
C...........................................................................
C.                                                                         .
```

```
C.                  ELEMENT MATRIX ASSEMBLY ROUTINE                   .
C.                                                                    .
C......................................................................
C.                                                                    .
C.    THE PURPOSE OF THIS SUBROUTINE IS TO ASSEMBLE THE ELEMENT       .
C.    COEFFICIENT MATRIX INTO THE SYSTEM COEFFICIENT MATRIX           .
C.    CONSISTENT WITH THE PROFILE SOLVER.                             .
C.                                                                    .
C......................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
      COMMON/IDATA/IEL,ISC,NEND,NEQ,NTMC,NUMEL,NUMNP,NUMSC
      COMMON/EQN/A(25,12),RHS(11,12),JD(11,12)
      COMMON/IELMNT/LD(3,10,12),NCON(3,10,12),NELTP(10,12)
      DIMENSION EFM(3,3)
C
C     LOOPING OVER THE COLUMNS.
C
C        WRITE(6,9001)(LD(I,IEL,ISC),I=1,NEND)
9001     FORMAT(' ADDSTF ',3I5)
      DO 20 J=1,NEND
         K = LD(J,IEL,ISC)
         IF (K .EQ. 0) GO TO 20
         L = JD(K,ISC) - K
C
C        LOOPING OVER THE ROWS.
C
         DO 10 I=1,NEND
            M = LD(I,IEL,ISC)
C
C           ASSEMBLING THE UPPER PROFILE COEFFICIENT MATRIX.
C
            IF (M .GT. K  .OR.  M .EQ. 0) GO TO 10
            M = L + M
            A(M,ISC) = A(M,ISC) + EFM(I,J)
10       CONTINUE
20    CONTINUE
C
      RETURN
      END
      SUBROUTINE CROUTE
C......................................................................
C.                                                                    .
C.                      CHANNEL ROUTING ROUTINE                       .
C.                                                                    .
C......................................................................
C.                                                                    .
C.    THE PURPOSE OF THIS SUBROUTINE TO PERFORM THE CHANNEL NETWORK   .
C.    ROUTING COMPUTATIONS BASED ON A DIFFUSION WAVE APPROXIMATION.   .
C.                                                                    .
C......................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
```

```
      COMMON/CDATA/CLGH,CSLP,CWTH,FCAR,RMANN,TRACF,YIN
      COMMON/DFILES/IF1,IF2
      COMMON/FILES/IIN1,INN2,IOUT
      COMMON/IDATA/IEL,ISC,NEND,NEQ,NTMC,NUMEL,NUMNP,NUMSC
      COMMON/ITDAT/ANORM,BCT,DWT,MBCIT,MDWIT,NBCS,NCSC
      COMMON/RLOG/AFAC,BACK
      COMMON/TIME/DT,RTH,RTH1,RTT,DELTA,IFDC,ITS,NFDC,NRAIN,NTS
      COMMON/EQN/A(25,12),RHS(11,12),JD(11,12)
      COMMON/FLOW/AFLW(12),DFLW(12),QFLW(12),QH(10)
      COMMON/ICHDAT/IDWIT(12),NLF(12),NEL(12),NH(12),NNP(12),NSLVC(5,12)
      COMMON/RCHDAT/CHR(12),CHS(12),CHW(12),CHZ(12),DSQ(12),WS(12)
      LOGICAL AFAC,BACK
C
2001  FORMAT(//14X,41H****  O U T F L O W   H Y D R O G R A P H,
     1 16H   D A T A  ****)
2002  FORMAT(/29X,'  RESULTS FOR TIME STEP:',I4
     1/20X,' NUMBER OF BOUNDARY CONDITION ITERATIONS:',I4
     2//18X,' CHANNEL  ITER    FLOWRATE      AREA        DEPTH',/)
2003  FORMAT(18X,I4,4X,I4,4X,1PD10.4,2X,1PD10.4,2X,1PD10.4)
C
      REWIND IF1
      REWIND IF2
C
C     INITIALIZING THE SOLUTION LOGICAL VARIABLES FOR
C     FORWARD ELIMINATION AND BACK SUBSTITUTION.
C
      AFAC = .FALSE.
      BACK = .TRUE.
C
C     CALCULATE THE INITIAL CHANNEL NETWORK FLOW DATA.
C
      CALL INSOL
C
C     DETERMINING THE TIME INTERPOLATED CHANNEL ELEMENT
C     LATERAL INFLOWS.
C
      CALL ELLFLW
C
C     LOOPING OVER THE TIME STEPS.
C
      DO 40 ITS=2,NTS
C
C        CALCULATING THE UPSTREAM FLOW RATE, LATERAL INFLOW FOR THE
C        CURRENT TIME STEP AND GENERATING THE PREDICTED FLOW VALUES.
C
         CALL CVAL
C
C        JUNCTION BOUNDARY CONDITION ITERATION LOOP.
C
         IBCIT = 0
10       IF (IBCIT .LT. MBCIT) THEN
            IBCIT = IBCIT + 1
            MAXIT = IBCIT*MDWIT
```

```
C
C           LOOPING OVER THE CHANNELS.
C
            DO 30 ISC=1,NUMSC
C
C              INITIALIZING THE CHANNEL VARIABLES.
C
               NUMEL = NEL(ISC)
               NUMNP = NNP(ISC)
               NEQ = NUMNP - 1
C
C              UPDATE UPSTREAM FLOW RATE FOR SLAVED CHANNEL?
C
               NMC = NSLVC(1,ISC) + 1
               IF (NMC .GT. 1) CALL UPSCF(NMC)
C
C              CHECK ON THE NUMBER OF CHANNEL ITERATIONS.
C
20             IF (IDWIT(ISC) .LT. MAXIT) THEN
                  IDWIT(ISC) = IDWIT(ISC) + 1
C
C                 CALCULATE THE NONLINEAR CHANNEL LOAD VECTOR.
C
                  CALL NRHS
C
C                 SOLVE FOR THE UNKNOWN NODAL FLOW AREAS.
C
                  CALL ACTCOL(A(1,ISC),JD(1,ISC))
C
C                 SOLUTION CONVERGENCE?
C
                  CALL CSOLN
                  IF (NCSC .EQ. 0) GO TO 20
               ENDIF
30          CONTINUE
C
C           CHECKING FOR DOWNSTREAM MASTER CHANNEL BOUNDARY CONDITION
C           CONVERGENCE.
C
            CALL DMCBC
            IF (NBCS .EQ. 0) GO TO 10
         ENDIF
C
C        UPDATING/OUTPUTTING THE CURRENT TIME STEP RESULTS.
C
         CALL OUTPUT(IBCIT)
40    CONTINUE
C
C     OUTPUTTING THE CHANNEL HYDROGRAPH RESULTS.
C
      REWIND IF2
      WRITE(IOUT,2001)
      DO 60 ITS=1,NTS
```

```
C
C        DISK FILE INPUT OF THE HYDROGRAPH DATA.
C
         READ(IF2) IBCIT,(IDWIT(ISC),AFLW(ISC),DFLW(ISC),QFLW(ISC),
     1                    ISC=1,NUMSC)
         WRITE(IOUT,2002) ITS,IBCIT
         DO 50 ISC=1,NUMSC
            WRITE(IOUT,2003) ISC,IDWIT(ISC),QFLW(ISC),AFLW(ISC),
     1                       DFLW(ISC)
50       CONTINUE
60    CONTINUE
C
      RETURN
      END
      SUBROUTINE INSOL
C.....................................................................
C.                                                                   .
C.                 INITIAL CHANNEL FLOW DATA ROUTINE                 .
C.                                                                   .
C.....................................................................
C.                                                                   .
C.   THE PURPOSES OF THIS SUBROUTINE ARE TO GENERATE THE INITIAL     .
C.   FLOW AREAS, DEPTHS AND FRICTION SLOPES FOR THE CHANNEL          .
C.   NETWORK USING PROFILE ANALYSIS.                                 .
C.                                                                   .
C.   DATA INPUT:                                                     .
C.                                                                   .
C.     YIN   - DOWNSTREAM DEPTH FOR THE LAST CHANNEL                 .
C.                                                                   .
C.....................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
      COMMON/CDATA/CLGH,CSLP,CWTH,FCAR,RMANN,TRACF,YIN
      COMMON/DFILES/IF1,IF2
      COMMON/FILES/IIN,INN2,IOUT
      COMMON/IDATA/IEL,ISC,NEND,NEQ,NTMC,NUMEL,NUMNP,NUMSC
      COMMON/ITDAT/ANORM,BCT,DWT,MBCIT,MDWIT,NBCS,NCSC
      COMMON/RDATA/ONE,R23,R35,R53,ZERO
      COMMON/TIME/DT,RTH,RTH1,RTT,DELTA,IFDC,ITS,NFDC,NRAIN,NTS
      COMMON/FLOW/AFLW(12),DFLW(12),QFLW(12),QH(10)
      COMMON/ICHDAT/IDWIT(12),NLF(12),NEL(12),NH(12),NNP(12),NSLVC(5,12)
      COMMON/RCHDAT/CHR(12),CHS(12),CHW(12),CHZ(12),DSQ(12),WS(12)
      COMMON/RELMNT/ELN(10,12),QLT1(10,12),QLT2(10,12)
      COMMON/RNODE/AT1(25,12),AT2(25,12),FDT(25,12),FRT(25,12),
     1             SF(25,12),XND(25,12)
      DIMENSION Y0(12)
C
1001  FORMAT(8F10.0)
2001  FORMAT(//18X,49H**** C O N V E R G E N C E   F A I L U R E ****)
2002  FORMAT(//19X,47H**** I N I T I A L   C O N D I T I O N S ****
     1//18X,'CHANNEL  NODE    FLOWRATE      AREA        DEPTH',/)
2003  FORMAT(18X,I4,4X,I4,4X,1PD10.4,2X,1PD10.4,3X,1PD10.4)
2004  FORMAT()
```

```
C
C     INITIALIZING THE NODAL FLOWS AND FRICTION SLOPES.
C
      DO 10 ISC=1,NUMSC
      DO 10 NOD=1,NNP(ISC)
         FRT(NOD,ISC) = QFLW(ISC)
         SF(NOD,ISC) = CHS(ISC)
10    CONTINUE
C
C     CALCULATE THE DOWNSTREAM CHANNEL NORMAL DEPTH?
C
      IF (YIN .LE. ZERO) THEN
C
C        CALCULATING THE DOWNSTREAM FLOW DEPTH USING SECANT
C        ITERATION ON MANNING'S FLOW AREA EQUATION.
C
         NUMNP = NNP(NUMSC)
         CON = DSQRT(SF(NUMNP,NUMSC))/CHR(NUMSC)
         ANM = ZERO
         P23 = (CHW(NUMSC) + 2.*FDT(NUMNP,NUMSC)*DSQ(NUMSC))**R23
         AN = (0.5*QFLW(NUMSC)*P23/CON)**R35
         FANM = QFLW(NUMSC)
         FAN =  0.5*QFLW(NUMSC)
C
C        CALCULATING THE UPDATED FLOW AREA AND DEPTH.
C
         ITER = 0
20       IF (ITER .LT. MDWIT) THEN
            ITER = ITER + 1
            ANP = AN - FAN*(AN - ANM)/(FAN - FANM)
            WSQ = CHW(NUMSC) + DSQRT(WS(NUMSC) + 4.*CHZ(NUMSC)*ANP)
            YIN = 2.*ANP/WSQ
C
C           CHECK FOR CONVERGENCE.
C
            IF (DABS(ANP - AN)/ANP .GT. DWT) THEN
               ANM = AN
               AN = ANP
               P23 = (CHW(NUMSC) + 2.*YIN*DSQ(NUMSC))**R23
               A53 = AN**R53
               FANM = FAN
               FAN = QFLW(NUMSC) - CON*A53/P23
               GO TO 20
            ENDIF
         ELSE
C
C           NONCONVERGED SOLUTION.
C
            WRITE(IOUT,2001)
            STOP
         ENDIF
      ENDIF
C.....................................................................
```

```
C.                                                                   .
C.                  P R O F I L E   A N A L Y S I S                  .
C.                                                                   .
C.....................................................................
C
      ANORM = ZERO
      YO(NUMSC) = YIN
C
C
C     LOOPING OVER THE CHANNELS.
C
      WRITE(IOUT,2002)
      DO 70 IMC=1,NUMSC
C
C        ESTABLISHING THE CHANNEL PARAMETERS.
C
         ISC = NUMSC - IMC + 1
         NUMNP = NNP(ISC)
         NMC = NSLVC(1,ISC) + 1
C
         C1 = QFLW(ISC)*QFLW(ISC)/64.348
         C2 = 0.5*QFLW(ISC)*QFLW(ISC)*CHR(ISC)*CHR(ISC)
C
C        INITIAL FLOW VALUES.
C
         Y1 = YO(ISC)
         FDT(NUMNP,ISC) = Y1
         AT1(NUMNP,ISC) = (CHW(ISC) + CHZ(ISC)*Y1)*Y1
C
C        CALCULATING THE NODAL FLOW DEPTHS.
C
         NUMNP1 = NUMNP - 1
         DO 40 NOD=1,NUMNP1
            NODE = NUMNP - NOD
            RL = ELN(NODE,ISC)
            C3 = C2*RL
            C4 = CHS(ISC)*RL
C
C           CALCULATING THE DOWNSTREAM FIXED VALUES.
C
            CALL CHANC(Y1,A2,HR43)
            C5 = Y1 + C1/A2 + C3/A2/HR43 - C4
C
C           GENERATE FIRST TRIAL DEPTH FOR SECANT METHOD (DEPTHS
C           EQUALS 1/3 OF DOWNSTREAM NODAL DEPTH).
C
            Y21 = 0.80*Y1
            CALL CHANC(Y21,A2,HR43)
            F21 = Y21 + C1/A2 - C3/A2/HR43 - C5
C
C           GENERATE SECOND TRIAL DEPTH FOR SECANT METHOD (DEPTH
C           EQUALS 2/3 OF DOWNSTREAM NODAL DEPTH).
C
```

```
            Y22 = 0.90*Y1
            CALL CHANC(Y22,A2,HR43)
            F22 = Y22 + C1/A2 - C3/A2/HR43 - C5
C
C           SECANT ITERATION FOR NODAL DEPTH.
C
            IF (DABS(F22-F21) .LT. DWT) F22 = ZERO
30          Y2 = Y22 - F22*(Y22 - Y21)/(F22 - F21)
            if(y2.le.zero)y2=0.5*y22
C
C           CHECK FOR CONVERGENCE.
C
            IF (DABS(Y2-Y22)/Y2 .GT. DWT) THEN
C
C              UPDATE THE NODAL ITERATION FLOW DEPTHS.
C
               Y21 = Y22
               F21 = F22
               Y22 = Y2
               CALL CHANC(Y22,A2,HR43)
               F22 = Y22 + C1/A2 - C3/A2/HR43 - C5
               GO TO 30
            ELSE
C
C              CONVERGED SOLUTION.
C
               Y1 = Y2
               FDT(NODE,ISC) = Y1
               AT1(NODE,ISC) = (CHW(ISC) + CHZ(ISC)*Y1)*Y1
            ENDIF
40       CONTINUE
C
C        UPDATING THE MASTER CHANNEL DOWNSTREAM FLOW DEPTHS.
C
         IF (NMC .GT. 1) THEN
            DO 50 I=2,NMC
               YO(NSLVC(I,ISC)) = Y1
50          CONTINUE
         ENDIF
C
C        OUTPUTTING THE INITIAL CHANNEL CONDITIONS AND CALCULATING
C        THE INITIAL FLOW AREA TWO-NORM.
C
         DO 60 NOD=1,NUMNP
            WRITE(IOUT,2003) ISC,NOD,FRT(NOD,ISC),AT1(NOD,ISC),
     1                       FDT(NOD,ISC)
            WRITE(6,2003) ISC,NOD,FRT(NOD,ISC),AT1(NOD,ISC),
     1                       FDT(NOD,ISC)
            ANORM = ANORM + AT1(NOD,ISC)*AT1(NOD,ISC)
60       CONTINUE
         WRITE(IOUT,2004)
C
         IDWIT(ISC) = 0
```

```
         AFLW(ISC) = AT1(NUMNP,ISC)
         DFLW(ISC) = FDT(NUMNP,ISC)
         QFLW(ISC) = FRT(NUMNP,ISC)
70    CONTINUE
C
C     DISK FILE STORAGE OF THE CHANNEL FLOW CALCULATIONS.
C
      WRITE(IF2) IBCIT,(IDWIT(ISC),AFLW(ISC),DFLW(ISC),QFLW(ISC),
     1                   ISC=1,NUMSC)
C
      RETURN
      END
      SUBROUTINE CHANC(Y,A,R)
C.......................................................................
C.                                                                     .
C.                   CHANNEL CHARACTERISTICS ROUTINE                   .
C.                                                                     .
C.......................................................................
C.                                                                     .
C.    THE PURPOSE OF THIS SUBROUTINE IS TO DETERMINE THE VALUES OF     .
C.    AREA SQUARED AND THE HYDRAULIC RADIUS RAISED TO THE 4/3'RDS      .
C.    POWER FOR A GIVEN FLOW DEPTH.                                    .
C.                                                                     .
C.......................................................................
C.
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
      COMMON/IDATA/IEL,ISC,NEND,NEQ,NTMC,NUMEL,NUMNP,NUMSC
      COMMON/RDATA/ONE,R23,R35,R53,ZERO
      COMMON/RCHDAT/CHR(12),CHS(12),CHW(12),CHZ(12),DSQ(12),WS(12)
C
C     CALCULATING FLOW AREA, WETTED PERIMETER, FLOW AREA SQUARED
C     AND HYDRAULIC RADIUS RAISED TO THE 4/3'RDS POWER.
C
      A1 = (CHW(ISC) + CHZ(ISC)*Y)*Y
      P = CHW(ISC) + 2.*Y*DSQ(ISC)
      A = A1*A1
      R = (A/P/P)**R23
C
      RETURN
      END
      SUBROUTINE ELLFLW
C.......................................................................
C.                                                                     .
C.                 ELEMENT LATERAL FLOW DATA ROUTINE                   .
C.                                                                     .
C.......................................................................
C.                                                                     .
C.    THE PURPOSE OF THIS SUBROUTINE IS TO CALCULATE THE ELEMENT       .
C.    LATERAL INFLOW DATA CONSISTENT WITH THE CURRENT TIME INCRE-      .
C.    MENT DATA; ACTUAL LATERAL INFLOW DATA IS OBTAINED FROM THE       .
C.    OVERLAND FLOW PROGRAM.                                           .
C.                                                                     .
C.......................................................................
```

```
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
      COMMON/FILES/IIN,INN2,IOUT
      COMMON/IDATA/IEL,ISC,NEND,NEQ,NTMC,NUMEL,NUMNP,NUMSC
      COMMON/RDATA/ONE,R23,R35,R53,ZERO
      COMMON/TIME/DT,RTH,RTH1,RTT,DELTA,IFDC,ITS,NFDC,NRAIN,NTS
      COMMON/ICHDAT/IDWIT(12),NLF(12),NEL(12),NH(12),NNP(12),NSLVC(5,12)
      COMMON/LFILES/IFL(12)
      COMMON/RELMNT/ELN(10,12),QLT1(10,12),QLT2(10,12)
      COMMON/TMPDAT/Q1(100),Q2(10,250)
C
C     LOOPING OVER THE CHANNELS.
C
      DO 70 ISC=1,NUMSC
C
         NUMEL = NEL(ISC)
C
C           DISK FILE INPUT OF THE LATERAL INFLOW DATA.
C
C        READ(IFL(ISC),REC=ITS) (QLT1(IEL,ISC),IEL=1,NUMEL)
         READ(IFL(ISC))(QLT1(IEL,ISC),IEL=1,NUMEL)
C
70    CONTINUE
C
      RETURN
      END
      SUBROUTINE CVAL
C.............................................................
C.                                                           .
C.               CURRENT TIME STEP CALCULATION ROUTINE       .
C.                                                           .
C.............................................................
C.                                                           .
C.    THE PURPOSES OF THIS SUBROUTINE ARE TO CALCULATE THE CURRENT   .
C.    TIME STEP LATERAL INFLOWS, PREDICTOR FLOW AREAS AND DEPTHS    .
C.    USING AN EXPLICIT FINITE DIFFERENCE ALGORITHM AND THE FIRST   .
C.    ITERATION FRICTION SLOPES AND FLOW RATES.                     .
C.                                                           .
C.............................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
      COMMON/DFILES/IF1,IF2
      COMMON/IDATA/IEL,ISC,NEND,NEQ,NTMC,NUMEL,NUMNP,NUMSC
      COMMON/ITDAT/ANORM,BCT,DWT,MBCIT,MDWIT,NBCS,NCSC
      COMMON/RDATA/ONE,R23,R35,R53,ZERO
      COMMON/TIME/DT,RTH,RTH1,RTT,DELTA,IFDC,ITS,NFDC,NRAIN,NTS
      COMMON/FLOW/AFLW(12),DFLW(12),QFLW(12),QH(10)
      COMMON/ICHDAT/IDWIT(12),NLF(12),NEL(12),NH(12),NNP(12),NSLVC(5,12)
      COMMON/IELMNT/LD(3,10,12),NCON(3,10,12),NELTP(10,12)
      COMMON/RCHDAT/CHR(12),CHS(12),CHW(12),CHZ(12),DSQ(12),WS(12)
      COMMON/RELMNT/ELN(10,12),QLT1(10,12),QLT2(10,12)
      COMMON/RNODE/AT1(25,12),AT2(25,12),FDT(25,12),FRT(25,12),
     1             SF(25,12),XND(25,12)
```

```
C
      DIMENSION QQ(10)
C
C     DISK FILE INPUT OF THE MASTER CHANNEL INFLOW RATES.
C
      READ(IF1) (QQ(I),I=1,NTMC)
C
      IC = ZERO
C
      DO 40 ISC=1,NUMSC
C
         NMC = NSLVC(1,ISC) + 1
         IF (NMC .NE. 1) GO TO 40
         IC = IC + 1
         QH(ISC)=QQ(IC)
C
40    CONTINUE
C
C     LOOPING OVER THE CHANNELS.
C
      DO 30 ISC=1,NUMSC
C
C        INITIALIZING THE CHANNEL VARIABLES.
C
         NUMEL = NEL(ISC)
         NUMNP = NNP(ISC)
         NEQ = NUMNP - 1
C
C        CALCULATING THE CURRENT TIME STEP LINEAR RIGHT HAND
C        SIDE VECTOR.
C
         CALL LRHS
C
C        CALCULATING THE NODAL FLOW AREAS AND DEPTHS AT TIME T +
C        THETA*DT USING AN EXPLICIT FINITE DIFFERENCE ALGORITHM.
C
         DO 20 IEL=1,NUMEL
            NEND = NELTP(IEL,ISC) + 1
            DQLT = RTT*QLT1(IEL,ISC)
            DQLT = ZERO
            DXT = RTT*NELTP(IEL,ISC)/ELN(IEL,ISC)
            DO 10 NOD=2,NEND
               NOD2 = NCON(NOD,IEL,ISC)
               NOD1 = NOD2 - 1
               AT2(NOD2,ISC) = DQLT + AT1(NOD2,ISC) - DXT*(FRT(NOD2,ISC)
     1                        - FRT(NOD1,ISC))
C
               IF(AT2(NOD2,ISC).LE.ZERO)AT2(NOD2,ISC)=0.5*AT1(NOD2,ISC)
C
C              CALCULATING THE PREDICTOR FLOW DEPTH BY SOLVING
C              THE QUADRATIC EQUATION:
C
C                  2                                         1/2
```

```
C             S  h  + wh - A = 0;  h = 2A/[w + (w*w + 4*S A)    ]
C              s                                         s
C
C             h is the positive root of the quadratic equation.
C
C
              WSQ = CHW(ISC) + DSQRT(WS(ISC)+4.*CHZ(ISC)*AT2(NOD2,ISC))
              FDT(NOD2,ISC) = 2.*AT2(NOD2,ISC)/WSQ
C
10          CONTINUE
20        CONTINUE
C
C         CALCULATING THE UPSTREAM FLOW AREA AND DEPTH
C         AT TIME T + THETA*DT.
C
          QLT1(1,ISC)=ZERO
          AT2(1,ISC) = RTT*QLT1(1,ISC) + AT1(1,ISC) + RTT*NELTP(1,ISC)
     1                 *(FRT(1,ISC) - FRT(2,ISC))/ELN(1,ISC)
C
           IF(AT2(1,ISC).LE.ZERO)AT2(1,ISC)=0.5*AT1(1,ISC)
C
          WSQ = CHW(ISC) + DSQRT(WS(ISC) + 4.*CHZ(ISC)*AT2(1,ISC))
          FDT(1,ISC) = 2.*AT2(1,ISC)/WSQ
C
C         CALCULATING THE UPSTREAM FLOW RATE AT TIME T + THETA*DT
C         FOR A MASTER
C
          NMC = NSLVC(1,ISC) + 1
          FFF = FRT (1,ISC)
          IF (NMC .EQ. 1) FRT(1,ISC) = RTH1*FRT(1,ISC) + RTH*QH(ISC)
C
C         GENERATING THE NODAL FRICTION SLOPES AND FLOW RATES.
C
          CALL FSFR
30      CONTINUE
C
C       ADJUSTING THE DOWNSTREAM MASTER CHANNEL BOUNDARY CONDITIONS.
C
        CALL DMCBC
C
        RETURN
        END
        SUBROUTINE LRHS
C.....................................................................
C.                                                                   .
C.                 LINEAR RIGHT HAND SIDE VECTOR ROUTINE             .
C.                                                                   .
C.....................................................................
C.                                                                   .
C.      THE PURPOSES OF THIS SUBROUTINE ARE TO CALCULATE THE CURRENT .
C.      TIME STEP LINEAR RIGHT HAND SIDE VECTOR.                     .
C.                                                                   .
C.....................................................................
```

```
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
      COMMON/IDATA/IEL,ISC,NEND,NEQ,NTMC,NUMEL,NUMNP,NUMSC
      COMMON/RDATA/ONE,R23,R35,R53,ZERO
      COMMON/TIME/DT,RTH,RTH1,RTT,DELTA,IFDC,ITS,NFDC,NRAIN,NTS
      COMMON/EQN/A(25,12),RHS(11,12),JD(11,12)
      COMMON/FLWMAT/AFM(3,3,2),QFV(3,2),VFM(3,3,2)
      COMMON/IELMNT/LD(3,10,12),NCON(3,10,12),NELTP(10,12)
      COMMON/LFILES/IFL(12)
      COMMON/RELMNT/ELN(10,12),QLT1(10,12),QLT2(10,12)
      COMMON/RNODE/AT1(25,12),AT2(25,12),FDT(25,12),FRT(25,12),
     1             SF(25,12),XND(25,12)
      DIMENSION P(3)
C
C     INITIALIZING THE LINEAR RIGHT HAND SIDE VECTOR.
C
      DO 10 I=1,NEQ
         RHS(I,ISC) = ZERO
10    CONTINUE
C
C     DISK FILE INPUT OF THE LATERAL INFLOWS AT TIME T AND TIME
C     T + DT.
C
C     READ(IFL(ISC),REC=ITS) (QLT2(IEL,ISC),IEL=1,NUMEL)
      READ(IFL(ISC)) (QLT2(IEL,ISC),IEL=1,NUMEL)
C
C     LOOPING OVER THE ELEMENTS.
C
      DO 40 IEL=1,NUMEL
         IELTP = NELTP(IEL,ISC)
         NEND = IELTP + 1
         RL = ELN(IEL,ISC)
C
C        CALCULATING THE ELEMENT LATERAL INFLOW AT TIME T + THETA*DT.
C
         QLT = RL*RTT*(RTH*QLT2(IEL,ISC) + RTH1*QLT1(IEL,ISC))
C
C        CALCULATING THE ELEMENT RIGHT HAND SIDE VECTOR.
C
         DO 20 I=1,NEND
            P(I) = QFV(I,IELTP)*QLT
            DO 20 J=1,NEND
               P(I) = P(I) + RL*AFM(I,J,IELTP)*AT1(NCON(J,IEL,ISC),ISC)
20       CONTINUE
C
C        ASSEMBLING THE ELEMENT RIGHT HAND SIDE VECTOR
C        INTO THE SYSTEM LINEAR RIGHT HAND SIDE VECTOR.
C
         DO 30 NOD=1,NEND
            IDF = LD(NOD,IEL,ISC)
            IF (IDF.GT. 0) RHS(IDF,ISC) = RHS(IDF,ISC) + P(NOD)
30       CONTINUE
40    CONTINUE
```

```
C
      RETURN
      END
      SUBROUTINE UPSCF(NMC)
C....................................................................
C.                                                                  .
C.                   UPSTREAM SLAVE CHANNEL FLOW ROUTINE            .
C.                                                                  .
C....................................................................
C.                                                                  .
C.    THE PURPOSE OF THIS SUBROUTINE IS TO CALCULATE THE SLAVED     .
C.    CHANNEL UPSTREAM FLOW RATE.                                   .
C.                                                                  .
C....................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
      COMMON/IDATA/IEL,ISC,NEND,NEQ,NTMC,NUMEL,NUMNP,NUMSC
      COMMON/RDATA/ONE,R23,R35,R53,ZERO
      COMMON/ICHDAT/IDWIT(12),NLF(12),NEL(12),NH(12),NNP(12),NSLVC(5,12)
      COMMON/RNODE/AT1(25,12),AT2(25,12),FDT(25,12),FRT(25,12),
     1             SF(25,12),XND(25,12)
C
C     ACCUMULATING THE SLAVED CHANNEL UPSTREAM FLOW RATE.
C
      TFR = ZERO
      DO 10 IC=2,NMC
         MCN = NSLVC(IC,ISC)
         TFR = TFR + FRT(NNP(MCN),MCN)
10    CONTINUE
      FRT(1,ISC) = TFR
C
      RETURN
      END
      SUBROUTINE NRHS
C....................................................................
C.                                                                  .
C.                NONLINEAR RIGHT HAND SIDE VECTOR ROUTINE          .
C.                                                                  .
C....................................................................
C.                                                                  .
C.    THE PURPOSE OF THIS SUBROUTINE IS TO CALCULATE THE NONLINEAR  .
C.    CHANNEL LOAD VECTOR.                                          .
C.                                                                  .
C....................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
      COMMON/IDATA/IEL,ISC,NEND,NEQ,NTMC,NUMEL,NUMNP,NUMSC
      COMMON/RDATA/ONE,R23,R35,R53,ZERO
      COMMON/TIME/DT,RTH,RTH1,RTT,DELTA,IFDC,ITS,NFDC,NRAIN,NTS
      COMMON/EQN/A(25,12),RHS(11,12),JD(11,12)
      COMMON/FLWMAT/AFM(3,3,2),QFV(3,2),VFM(3,3,2)
      COMMON/IELMNT/LD(3,10,12),NCON(3,10,12),NELTP(10,12)
      COMMON/RELMNT/ELN(10,12),QLT1(10,12),QLT2(10,12)
```

```
      COMMON/RLOAD/B(11)
      COMMON/RNODE/AT1(25,12),AT2(25,12),FDT(25,12),FRT(25,12),
     1              SF(25,12),XND(25,12)
      DIMENSION P(3)
C
C     INITIALIZING THE NONLINEAR RIGHT HAND SIDE VECTOR.
C
      DO 10 I=1,NEQ
         B(I) = RHS(I,ISC)
10    CONTINUE
C
C     LOOPING OVER THE ELEMENTS.
C
      DO 40 IEL=1,NUMEL
         IELTP = NELTP(IEL,ISC)
         NEND = IELTP + 1
C
C        CALCULATING THE ELEMENT NONLINEAR RIGHT HAND SIDE VECTOR.
C
         DO 20 I=1,NEND
            P(I) = ZERO
            DO 20 J=1,NEND
               P(I) = P(I) - VFM(I,J,IELTP)*FRT(NCON(J,IEL,ISC),ISC)
20       CONTINUE
C
C        ASSEMBLING THE CURRENT ELEMENT FLOW VECTOR INTO THE
C        SYSTEM RIGHT HAND SIDE VECTOR.
C
         DO 30 NOD=1,NEND
            IDF = LD(NOD,IEL,ISC)
            IF (IDF .GT. 0) B(IDF) = B(IDF) + RTT*P(NOD)
30       CONTINUE
40    CONTINUE
C
C     ASSEMBLING THE BOUNDARY CONDITION INTO THE RIGHT HAND
C     SIDE VECTOR.
C
      IELTP = NELTP(NUMEL,ISC)
      NEND = IELTP + 1
      RL = ELN(NUMEL,ISC)
      DO 50 I=1,IELTP
         B(LD(I,NUMEL,ISC)) = B(LD(I,NUMEL,ISC)) - RL*AFM(I,NEND,IELTP)
     1                        *AT2(NCON(NEND,NUMEL,ISC),ISC)
50    CONTINUE
C
      RETURN
      END
      SUBROUTINE ACTCOL(A,JDIAG)
C.....................................................................
C.                                                                   .
C.            SYMMETRIC PROFILE EQUATION SOLUTION ROUTINE            .
C.                                                                   .
C.....................................................................
```

```
C.                                                                   .
C.    THE PURPOSES OF THIS SUBROUTINE ARE TO PERFORM FORWARD ELIMI-  .
C.    NATION AND BACK SUBSTITUTION OPERATIONS ON A SYMMETRIC         .
C.    PROFILE MATRIX USING GAUSS-CROUT ELIMINATION.                  .
C.                                                                   .
C.....................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
      COMMON/IDATA/IEL,ISC,NEND,NEQ,NTMC,NUMEL,NUMNP,NUMSC
      COMMON/RDATA/ONE,R23,R35,R53,ZERO
      COMMON/RLOG/AFAC,BACK
      COMMON/RLOAD/B(11)
      DIMENSION A(25),JDIAG(11)
      LOGICAL AFAC,BACK
C
C                                             T
C     FACTOR THE COEFFICIENT MATRIX A INTO (U)   * D * U AND REDUCE THE
C     THE RIGHT HAND SIDE VECTOR B.
C
      JR = 0
      DO 600 J=1,NEQ
         JD = JDIAG(J)
         JH = JD - JR
         IS = J - JH + 2
         IF (JH-2) 600,300,100
100      IF (.NOT. AFAC) GO TO 500
         IE = J - 1
         K = JR + 2
         ID = JDIAG(IS-1)
C
C        REDUCE ALL EQUATIONS EXCEPT THE DIAGONAL.
C
         DO 200 I=IS,IE
            IR = ID
            ID = JDIAG(I)
            IH = MIN0(ID-IR-1,I-IS+1)
            IF (IH .GT. 0) A(K) = A(K) - DOT(A(K-IH),A(ID-IH),IH)
            K = K + 1
200      CONTINUE
C
C        REDUCE THE DIAGONAL TERM.
C
300      IF (.NOT. AFAC) GO TO 500
         IR = JR + 1
         IE = JD - 1
         K = J - JD
         DO 400 I=IR,IE
            ID = JDIAG(K+I)
            IF (A(ID) .EQ. ZERO) GO TO 400
            D = A(I)
            A(I) = A(I)/A(ID)
            A(JD) = A(JD) - D*A(I)
400      CONTINUE
```

```
C
C        REDUCE THE RIGHT HAND SIDE VECTOR?
C
500      IF (BACK) B(J) = B(J) - DOT(A(JR+1),B(IS-1),JH-1)
600   JR = JD
      IF (.NOT. BACK) RETURN
C
C     DIVIDE BY THE DIAGONAL PIVOTS.
C
      DO 700 I=1,NEQ
         ID = JDIAG(I)
         IF (A(ID) .NE. ZERO) B(I) = B(I)/A(ID)
700   CONTINUE
C
C     BACK SUBSTITUTION.
C
      J = NEQ
      JD = JDIAG(J)
800   D = B(J)
         J = J - 1
         IF (J .LE. 0) RETURN
         JR = JDIAG(J)
         IF (JD-JR .LE. 1) GO TO 1000
         IS = J - JD + JR + 2
         K = JR - IS + 1
         DO 900 I=IS,J
            B(I) = B(I) - D*A(I+K)
900      CONTINUE
1000     JD = JR
      GO TO 800
C
      END
      FUNCTION DOT(A,B,N)
C.....................................................................
C.                                                                   .
C.                  DOT PRODUCT FUNCTION SUBPROGRAM                  .
C.                                                                   .
C.....................................................................
C.                                                                   .
C.    THE PURPOSE OF THIS SUBPROGRAM IS TO EVALUATE VECTOR DOT       .
C.    PRODUCTS.                                                      .
C.                                                                   .
C.....................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
      COMMON/RDATA/ONE,R23,R35,R53,ZERO
      DIMENSION A(N),B(N)
C
C     EVALUATING THE VECTOR DOT PRODUCT.
C
      DOT = ZERO
      DO 10 I=1,N
         DOT = DOT + A(I)*B(I)
```

```
10    CONTINUE
C
      RETURN
      END
      SUBROUTINE CSOLN
C.....................................................................
C.                                                                   .
C.                  SOLUTION CONVERGENCE CHECK ROUTINE               .
C.                                                                   .
C.....................................................................
C.                                                                   .
C.    THE PURPOSE OF THIS SUBROUTINE IS TO DETERMINE SOLUTION        .
C.    CONVERGENCE.                                                   .
C.                                                                   .
C.....................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
      COMMON/IDATA/IEL,ISC,NEND,NEQ,NTMC,NUMEL,NUMNP,NUMSC
      COMMON/ITDAT/ANORM,BCT,DWT,MBCIT,MDWIT,NBCS,NCSC
      COMMON/RDATA/ONE,R23,R35,R53,ZERO
      COMMON/TIME/DT,RTH,RTH1,RTT,DELTA,IFDC,ITS,NFDC,NRAIN,NTS
      COMMON/ICHDAT/IDWIT(12),NLF(12),NEL(12),NH(12),NNP(12),NSLVC(5,12)
      COMMON/RCHDAT/CHR(12),CHS(12),CHW(12),CHZ(12),DSQ(12),WS(12)
      COMMON/RLOAD/B(11)
      COMMON/RNODE/AT1(25,12),AT2(25,12),FDT(25,12),FRT(25,12),
     1             SF(25,12),XND(25,12)
C
C     CALCULATING THE INCREMENTAL FLOW AREA TWO-NORM AND UPDATING
C     THE CURRENT TIME STEP NODAL FLOW AREA AND DEPTH VALUES.
C
      ATN = ZERO
      DO 10 NOD=1,NEQ
         ADF = B(NOD) - AT2(NOD,ISC)
         ATN = ATN + ADF*ADF
         AT2(NOD,ISC) = B(NOD)
C
         IF(AT2(NOD,ISC).LE.ZERO)AT2(NOD,ISC)=0.5*AT1(NOD,ISC)
C
         WSQ = CHW(ISC) + DSQRT(WS(ISC) + 4.*CHZ(ISC)*AT2(NOD,ISC))
         FDT(NOD,ISC) = 2.*AT2(NOD,ISC)/WSQ
10    CONTINUE
C
C     GENERATING THE NODAL FRICTION SLOPES AND FLOW RATES.
C
      CALL FSFR
C
C     SOLUTION CONVERGENCE?
C
      NCSC = 0
      IF (ATN/ANORM .LT. DWT) NCSC = 1
C
      RETURN
      END
```

```
      SUBROUTINE DMCBC
C.....................................................................
C.                                                                   .
C.        DOWNSTREAM MASTER CHANNEL BOUNDARY CONDITION ROUTINE       .
C.                                                                   .
C.....................................................................
C.                                                                   .
C.    THE PURPOSE OF THIS SUBROUTINE IS TO CALCULATE THE AVERAGE     .
C.    FLOW DEPTH AT THE MASTER-SLAVE JUNCTION(S) AND DETERMINE       .
C.    IF THE DOWNSTREAM MASTER CHANNEL BOUNDARY CONDITION HAS        .
C.    CONVERGED.                                                     .
C.                                                                   .
C.....................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
      COMMON/IDATA/IEL,MCN,NEND,NEQ,NTMC,NUMEL,NUMNP,NUMSC
      COMMON/ITDAT/ANORM,BCT,DWT,MBCIT,MDWIT,NBCS,NCSC
      COMMON/RDATA/ONE,R23,R35,R53,ZERO
      COMMON/TIME/DT,RTH,RTH1,RTT,DELTA,IFDC,ITS,NFDC,NRAIN,NTS
      COMMON/ICHDAT/IDWIT(12),NLF(12),NEL(12),NH(12),NNP(12),NSLVC(5,12)
      COMMON/IELMNT/LD(3,10,12),NCON(3,10,12),NELTP(10,12)
      COMMON/RCHDAT/CHR(12),CHS(12),CHW(12),CHZ(12),DSQ(12),WS(12)
      COMMON/RNODE/AT1(25,12),AT2(25,12),FDT(25,12),FRT(25,12),
     1             SF(25,12),XND(25,12)
C
C     INITIALIZING THE SUBROUTINE PARAMETERS.
C
      NBCS = 0
      NMBC = 0
      NSLC = NUMSC - NTMC
      IBSC = NTMC + 1
C
C     SINGLE CHANNEL ANALYSIS?
C
      IF (NSLC .EQ. 0) THEN
         NBCS = 1
         RETURN
      ENDIF
C
C     LOOPING OVER THE SLAVED CHANNELS.
C
      DO 20 ISC=IBSC,NUMSC
         NMC = NSLVC(1,ISC) + 1
C
C        ACCUMULATING THE CHANNEL JUNCTION FLOW DEPTH.
C        CALCULATING THE MASTER CHANNEL FLOW AREAS.
C        SETTING THE MASTER CHANNEL FLOW DEPTHS.
C
         SFD = FDT(1,ISC)
         TFD = SFD
         TFR = ZERO
         DO 10 IC=2,NMC
            MCN = NSLVC(IC,ISC)
```

```
            TFD = TFD + FDT(NNP(MCN),MCN)
            AT2(NNP(MCN),MCN) = (CHW(MCN) + CHZ(MCN)*SFD)*SFD
            FDT(NNP(MCN),MCN) = SFD
10       CONTINUE
C
C        CHECK FOR CHANNEL NETWORK JUNCTION CONVERGENCE.
C
         AFD = TFD/NMC
         IF (DABS(SFD - AFD)/SFD .LT. BCT) NMBC = NMBC + 1
20    CONTINUE
C
C     ESTABLISHING THE CHANNEL NETWORK CONVERGENCE CONDITION.
C
      NBCS = NMBC/NSLC
C
      RETURN
      END
      SUBROUTINE OUTPUT(IBCIT)
C.....................................................................
C.                                                                   .
C.                   FLOW PARAMETER UPDATE ROUTINE                   .
C.                                                                   .
C.....................................................................
C.                                                                   .
C.    THE PURPOSES OF THIS SUBROUTINE ARE TO UPDATE THE NODAL FLOW   .
C.    PARAMETERS AND OUTPUT THE FLOW DEPTH PROFILE.                  .
C.                                                                   .
C.....................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
      COMMON/DFILES/IF1,IF2
      COMMON/FILES/IIN,INN2,IOUT
      COMMON/IDATA/IEL,ISC,NEND,NEQ,NTMC,NUMEL,NUMNP,NUMSC
      COMMON/ITDAT/ANORM,BCT,DWT,MBCIT,MDWIT,NBCS,NCSC
      COMMON/TIME/DT,RTH,RTH1,RTT,DELTA,IFDC,ITS,NFDC,NRAIN,NTS
      COMMON/FLOW/AFLW(12),DFLW(12),QFLW(12),QH(10)
      COMMON/ICHDAT/IDWIT(12),NLF(12),NEL(12),NH(12),NNP(12),NSLVC(5,12)
      COMMON/RCHDAT/CHR(12),CHS(12),CHW(12),CHZ(12),DSQ(12),WS(12)
      COMMON/RELMNT/ELN(10,12),QLT1(10,12),QLT2(10,12)
      COMMON/RNODE/AT1(25,12),AT2(25,12),FDT(25,12),FRT(25,12),
     1             SF(25,12),XND(25,12)
      DIMENSION ITER(12)
C
2001  FORMAT(///23X,'  THE NODAL FLOW DEPTHS FOR TIME: ',F7.2,
     1/22X,'  THE NUMBER OF BOUNDARY ITERATIONS: ',I6,
     2//10X,' CHANNEL  ITER   NODE   FLOWRATE   AREA   DEPTH  ',/)
2002  FORMAT(11X,I4,6X,I4,5X,I4,3X,1PD10.4,3X,1PD10.4,3X,1PD11.5)
2003  FORMAT()
C
C     UPDATING/CALCULATING THE CHANNEL DATA.
C
      ANORM = ZERO
      DO 40 ISC=1,NUMSC
```

```
C
C        UPDATING THE LATERAL INFLOWS.
C
         NUMEL = NEL(ISC)
         DO 10 IEL=1,NUMEL
            QLT1(IEL,ISC) = QLT2(IEL,ISC)
10       CONTINUE
C
C        UPDATING THE NODAL FLOW AREA AND DEPTH VECTORS; CALCULATING
C        THE FLOW AREA TWO-NORM.
C
         NUMNP = NNP(ISC)
         DO 20 NOD=1,NUMNP
            AT = (AT2(NOD,ISC) - RTH1*AT1(NOD,ISC))/RTH
C
            AT2(NOD,ISC) = AT
C
            IF(AT2(NOD,ISC).LE.ZERO)AT2(NOD,ISC)=0.5*AT1(NOD,ISC)
C
            AT1(NOD,ISC) = AT
C
            WSQ = CHW(ISC) + DSQRT(WS(ISC) + 4.*CHZ(ISC)*AT1(NOD,ISC))
            FDT(NOD,ISC) = 2.*AT1(NOD,ISC)/WSQ
            ANORM = ANORM + AT1(NOD,ISC)*AT1(NOD,ISC)
20       CONTINUE
C
C        CALCULATING THE UPSTREAM FLOW RATE AT TIME T + DT
C        FOR EITHER A MASTER OR SLAVED CHANNEL.
C
         NMC = NSLVC(1,ISC) + 1
         IF (NMC .EQ. 1) THEN
C
C           MASTER CHANNEL.
C
            FRT(1,ISC) = QH(ISC)
         ELSE
C
C           SLAVED CHANNEL.
C
            TFR = ZERO
            DO 30 IC=2,NMC
               MCN = NSLVC(IC,ISC)
               TFR = TFR + FRT(NNP(MCN),MCN)
30          CONTINUE
            FRT(1,ISC) = TFR
         ENDIF
C
C        UPDATING THE FRICTION SLOPES AND FLOW RATES.
C
         CALL FSFR
C
C        UPDATING THE DIFFUSION WAVE ITERATION COUNTER AND
C        THE OUTFLOW HYDROGRAPH VARIABLES.
```

```
C
         ITER(ISC) = IDWIT(ISC)
         IDWIT(ISC) = 0
         AFLW(ISC) = AT1(NUMNP,ISC)
         DFLW(ISC) = FDT(NUMNP,ISC)
         QFLW(ISC) = FRT(NUMNP,ISC)
40    CONTINUE
C
C     DISK FILE STORAGE OF THE CHANNEL FLOW CALCULATIONS.
C
      WRITE(IF2) IBCIT,(ITER(ISC),AFLW(ISC),DFLW(ISC),QFLW(ISC),
     1                  ISC=1,NUMSC)
C
C     CHECK FOR FLOW DEPTH OUTPUT.
C
      IFDC = IFDC + 1
      IF (IFDC .EQ. NFDC) THEN
         IFDC = 0
         STIME = (ITS - 1)*DT
         WRITE(IOUT,2001) STIME,IBCIT
         DO 60 ISC=1,NUMSC
C
C           OUTPUTTING THE FLOW DEPTH PROFILE.
C
            DO 50 NOD=1,NUMNP
               WRITE(IOUT,2002) ISC,ITER(ISC),NOD,FRT(NOD,ISC),
     1                          AT1(NOD,ISC),FDT(NOD,ISC)
50          CONTINUE
            WRITE(IOUT,2003)
60       CONTINUE
      ENDIF
C
      RETURN
      END
      SUBROUTINE FSFR
C.....................................................................
C.                                                                   .
C.              FRICTION SLOPE/FLOW RATE UPDATE ROUTINE              .
C.                                                                   .
C.....................................................................
C.                                                                   .
C.    THE PURPOSES OF THIS SUBROUTINE ARE TO UPDATE THE NODAL        .
C.    FRICTION SLOPES AND CALCULATE THE UNKNOWN NODAL FLOW RATES.    .
C.                                                                   .
C.....................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
      COMMON/IDATA/IEL,ISC,NEND,NEQ,NTMC,NUMEL,NUMNP,NUMSC
      COMMON/TIME/DT,RTH,RTH1,RTT,DELTA,IFDC,ITS,NFDC,NRAIN,NTS
      COMMON/RCHDAT/CHR(12),CHS(12),CHW(12),CHZ(12),DSQ(12),WS(12)
      COMMON/RNODE/AT1(25,12),AT2(25,12),FDT(25,12),FRT(25,12),
     1             SF(25,12),XND(25,12)
C
```

```
C     LOOPING OVER THE NODES.
C
      DO 10 NOD=2,NUMNP
         NDM = NOD - 1
         CALL FSLOPE(FS,NDM,NOD)
         SF(NOD,ISC) = FS
         CALL FRATE(FR,NOD)
         FRT(NOD,ISC) = FR
10    CONTINUE
C
      RETURN
      END
      SUBROUTINE FSLOPE(FS,NDM,NOD)
C.....................................................................
C.                                                                   .
C.             FRICTION SLOPE EVALUATION FUNCTION ROUTINE            .
C.                                                                   .
C.....................................................................
C.                                                                   .
C.    THE PURPOSE OF THIS SUBROUTINE IS TO EVALUATE THE FRICTION     .
C.    FRICTION SLOPE USING BACKWARD FINITE DIFFERENCES.              .
C.                                                                   .
C.....................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
      COMMON/IDATA/IEL,ISC,NEND,NEQ,NTMC,NUMEL,NUMNP,NUMSC
      COMMON/TIME/DT,RTH,RTH1,RTT,DELTA,IFDC,ITS,NFDC,NRAIN,NTS
      COMMON/RCHDAT/CHR(12),CHS(12),CHW(12),CHZ(12),DSQ(12),WS(12)
      COMMON/RELMNT/ELN(10,12),QLT1(10,12),QLT2(10,12)
      COMMON/RNODE/AT1(25,12),AT2(25,12),FDT(25,12),FRT(25,12),
     1             SF(25,12),XND(25,12)
C
C     CALCULATING THE NODAL DEPTH GRADIENT.
C
C     DDX = (FDT(NOD,ISC)-FDT(NDM,ISC))/(XND(NOD,ISC)-XND(NDM,ISC))
      DDX = (FDT(NOD,ISC)-FDT(NDM,ISC))/ELN(NDM,ISC)
C
C     CALCULATING THE FRICTION SLOPE; CHECK FOR NEGATIVE SLOPE.
C
      FS = CHS(ISC) - DDX
      IF (FS .LT. ZERO) FS = RTH1*SF(NOD,ISC)
C
      RETURN
      END
      SUBROUTINE FRATE(FR,NOD)
C.....................................................................
C.                                                                   .
C.                     FLOW RATE EVALUATION ROUTINE                  .
C.                                                                   .
C.....................................................................
C.                                                                   .
C.    THE PURPOSE OF THIS SUBROUTINE IS TO EVALUATE THE FLOW RATE.   .
C.                                                                   .
```

```
C.................................................................
C
      IMPLICIT REAL*8(A-H,O-Z), INTEGER*4(I-N)
      COMMON/IDATA/IEL,ISC,NEND,NEQ,NTMC,NUMEL,NUMNP,NUMSC
      COMMON/RDATA/ONE,R23,R35,R53,ZERO
      COMMON/RCHDAT/CHR(12),CHS(12),CHW(12),CHZ(12),DSQ(12),WS(12)
      COMMON/RNODE/AT1(25,12),AT2(25,12),FDT(25,12),FRT(25,12),
     1             SF(25,12),XND(25,12)
C
C     CALCULATING THE FLOW RATE.
C
      P23 = (CHW(ISC) + 2.*FDT(NOD,ISC)*DSQ(ISC))**R23
C
C     CALCULATE FLOWRATE USING MANNING'S EQUATION
C
      FR = (SF(NOD,ISC)**.5)*(AT2(NOD,ISC)**R53)/(CHR(ISC)*P23)
C
      RETURN
      END
```

APPENDIX E

CDATA and HDATA FILES FOR EXAMPLE RILL SYSTEM

Table E.1 CDATA File for Example Network

```
KENTUCY RILL NETWORK MODEL
PLOT S2R2    (SUBSOIL RUN)
    8    0   90    0    0    0
       0.0       0.0       0.0     60.00       2.0       .02
         3    1800.0
      4.35      4.35      4.35      4.35
      15.5       .28       .42       .19
      25.0      25.0      3.89      3.89       .52       .52
```

Table E.2 HDATA File for Example Network

```
9
1    9
     1          1          5  11.78560   0.08045   4.48515
0    0    0     0   0
1    1   30   296   0.15063   0.05927   0.17109  10.58602   4.48515   2.54000
1    2   29   394   0.17002   0.04156   0.14640  10.23766   4.16451   2.54000
2    1   30   395   0.12855   0.05054   0.18167   5.01476   4.16322   2.54000
2    2   24   492   0.38564   0.12544   0.09253   9.08366   3.65031   2.54000
3    1   24   493   0.17980   0.07235   0.09804   4.14928   3.64967   2.54000
3    2   26   591   0.16471   0.04156   0.15323   7.75602   3.30580   2.54000
4    1   26   592   0.22478   0.07005   0.11050   2.86814   3.30451   2.54000
4    2   22   686   0.34167   0.08315   0.07979   3.92599   2.73935   2.54000
5    1   22   687   0.16765   0.04789   0.11032   3.07073   2.73870   1.62560
5    2    9   747   1.14977   0.30272   0.09522   6.01680   1.42193   1.62560
6    0    9   747   1.42193   3.14960   0.06865  12.00394   1.42193   0.00000
     1          2          2   2.87020   0.09135   3.02451
0    0    0     0   0
1    1   30   296   0.48466   0.10211   0.32534  17.08169   3.02451   2.54000
1    2   38   394   1.20437   0.22834   0.13204  18.91173   1.33548   2.54000
2    1   36   395   0.00402   0.05292   0.05633   1.83722   1.33096   0.33020
2    2   36   406   0.04050   0.52493   0.10382  13.21119   1.29096   0.33020
3    0   36   406   1.29096   3.08779   0.07881   9.11308   1.29096   0.00000
     1          3          2   3.17500   0.08911   1.80774
0    0    0     0   0
1    1   25    99   0.21744   0.07061   0.17144   7.03517   1.80774   2.54000
1    2   25   198   0.18772   0.03556   0.29095   6.84882   1.40258   2.54000
2    1   25   199   0.05979   0.13818   0.31095   6.05172   1.40193   0.63500
2    2   27   223   0.04924   0.08839   0.15705   2.95836   1.29355   0.63500
3    0   27   223   1.29355   5.12656   0.10004  16.40262   1.29355   0.00000
     1          4          6  13.00480   0.09165   6.51998
0    0    0     0   0
1    1   19    98   0.27016   0.10414   0.20234   7.33167   6.51998   2.54000
1    2   17   197   0.28597   0.08484   0.14314   6.19629   5.96386   2.54000
2    1   17   198   0.28823   0.10730   0.15964   6.50636   5.95870   2.54000
2    2    8   295   0.43371   0.13659   0.10018   6.87874   5.24192   2.54000
3    1    8   296   1.09234   0.21590   0.08861   4.14036   5.23805   2.54000
3    2   13   395   0.65540   0.10414   0.08208   7.39363   3.49419   2.54000
4    1   13   396   0.43567   0.31370   0.09594   6.76520   3.49225   2.54000
4    2   11   492   0.18239   0.10605   0.10437   4.81643   2.87612   2.54000
5    1   11   493   0.74174   0.26365   0.08671   5.91189   2.87548   2.54000
5    2   11   592   0.79890   0.25857   0.07229   7.32300   1.33548   2.54000
6    1   11   593   0.01866   0.25188   0.06282   4.33356   1.33419   0.30480
6    2   11   604   0.02650   0.33232   0.07857   3.44384   1.29032   0.30480
7    0   11   604   1.29032   2.68556   0.08270   4.07609   1.29032   0.00000
     1          5          2   0.76200   0.09839   1.73096
0    0    0     0   0
1    1   11    24   0.02254   0.15784   0.05317  16.49235   1.73096   0.38100
1    2    9    37   0.00520   0.01270   0.13024   1.76756   1.70322   0.38100
2    1    9    38   0.03471   0.18687   0.08684   2.79127   1.69806   0.38100
2    2    6    51   0.01497   0.05261   0.35999   4.55832   1.65354   0.38100
3    0    6    51   1.65354   4.87094   0.11984  14.57252   1.65354   0.00000
```

```
     1         6         3   5.10540   0.08206   9.08514
2    1    2    0    0
1    1   26   99   0.04890   0.01426   0.24467   6.60282   9.08514   2.54000
1    2   33  196   0.32658   0.06920   0.24575   9.32319   8.70966   2.54000
2    1   33  197   0.41194   0.11058   0.31561  13.03680   8.70901   2.54000
2    2   30  295   0.78096   0.18319   0.25377  16.30974   7.51676   2.54000
3    1   30  296   0.00065   0.25400   3.19764 271.93651   7.51159   0.02540
3    2   30  296   0.00065   0.33020   1.07527  92.52106   7.51159   0.02540
     1         7         2   0.20320   0.04199  10.97804
2    6    3    0    0
1    1   20   98   0.00000   0.00000   1.07527   2.00000  10.97804   0.10160
1    2   22   99   0.00387   0.00000   0.07468   2.48997  10.97417   0.10160
2    1   25   98   0.04557   0.05080   0.04041   2.71021  10.90255   0.10160
2    2   25   99   0.02604   0.00000   2.01808 116.29612  10.90255   0.10160
     1         8         2   1.93040   0.07830  18.07286
2    7    4    0    0
1    1   11   24   0.09550   0.08155   0.13698   6.36241  18.07286   0.96520
1    2   16   61   0.25676   0.19451   0.38821  16.09749  17.72060   0.96520
2    1   16   62   0.04390   0.07963   0.19027   7.70721  17.71996   0.96520
2    2   19   98   0.11288   0.18329   0.27949  13.09665  17.56383   0.96520
     1         9         2   0.66040   0.10842  19.93608
2    8    5    0    0
1    1   18    1   0.02422   0.08011   0.27187   2.75743  19.93544   0.33020
1    2   15   13   0.03190   0.08011   0.32801   8.26235  19.87931   0.33020
2    1   15   14   0.02809   0.07620   0.15373   2.63955  19.86963   0.33020
2    2   11   24   0.04288   0.08775   0.62359  15.77852  19.80834   0.33020
2    7
     2         1         3   6.83260   0.07476   4.82386
0    0    0    0    0
1    1   50  509   0.21448   0.08432   0.11030   6.31251   4.82386   2.54000
1    2   59  605   0.29713   0.09086   0.13692   6.50199   4.31225   2.54000
2    1   59  606   0.53535   0.16166   0.08416   9.39020   4.31031   2.54000
2    2   67  701   0.66529   0.17357   0.08413   5.06645   3.11161   2.54000
3    1   67  702   0.76378   0.30592   0.13917   4.00178   3.11096   1.75260
3    2   66  769   1.04202   0.39034   0.11187   8.17127   1.30580   1.75260
4    0   66  769   1.30580   2.32255   0.09397   9.15547   1.30580   0.00000
     2         2         2   4.31800   0.07087   4.39160
0    0    0    0    0
1    1   50  509   0.87978   0.19345   0.13014   5.65531   4.39160   2.54000
1    2   48  607   0.51118   0.08672   0.08864   6.99686   3.00064   2.54000
2    1   47  608   1.02211   0.30365   0.10586   5.13272   2.99999   1.77800
2    2   51  673   0.66821   0.17972   0.09413   3.71031   1.31032   1.77800
3    0   51  673   1.31032   3.98042   0.09154  12.89723   1.31032   0.00000
     2         3         2   2.00660   0.07189   3.12773
0    0    0    0    0
1    1   63  391   0.11339   0.10160   0.11516  14.05265   3.12773   1.01600
1    2   74  421   0.22919   0.17370   0.18189  11.77137   2.78515   1.01600
2    1   74  422   0.44112   0.15891   0.12557   9.29863   2.78322   0.99060
2    2   74  460   1.03952   0.34909   0.11422   8.18986   1.30451   0.99060
3    0   74  460   1.30451   4.84293   0.07147   9.95920   1.30451   0.00000
     2         4         2   0.99060   0.09843   1.55225
0    0    0    0    0
1    1   44  141   0.06765   0.08001   0.10538   2.88442   1.55225   0.50800
```

```
   1   2   47  160   0.08912   0.08001   0.13046   2.08938   1.39548   0.50800
   2   1   48  161   0.05647   0.20743   0.10611   2.48336   1.39096   0.48260
   2   2   46  178   0.02547   0.06632   0.18808   5.16197   1.31355   0.48260
   3   0   46  178   1.31355   4.17002   0.10598  16.79567   1.31355   0.00000
       2         5         2   3.20040   0.07839  11.25223
   2   1    2    0    0
   1   1   63  391   0.71208   0.14652   0.16044  13.15183  11.25223   2.54000
   1   2   50  483   1.10727   0.20881   0.17564  12.02751   9.43288   2.54000
   2   1   50  484   0.05289   0.14752   0.29219   7.44666   9.42385   0.66040
   2   2   50  509   0.15937   0.42008   0.06722   8.99204   9.22062   0.66040
       2         6         3   6.50240   0.09367  17.16577
   2   5    3    0    0
   1   1   44  141   0.16245   0.02592   0.20980   9.32608  17.16577   2.54000
   1   2   55  237   1.23690   0.16916   0.10457   6.61626  15.76641   2.54000
   2   1   55  238   0.14607   0.07235   0.09523   5.09056  15.76577   2.54000
   2   2   59  336   1.05328   0.49722   0.13798  14.15973  14.56706   2.54000
   3   1   59  337   0.03168   0.08867   0.16202  12.17314  14.56642   1.42240
   3   2   63  391   0.13800   0.37638   0.13936  13.20108  14.39738   1.42240
       2         7         2   3.60680   0.08468  20.12253
   2   6    4    0    0
   1   1   39    1   0.58210   0.10871   0.08433  12.93163  20.12189   2.54000
   1   2   45  100   0.51274   0.07036   0.26195   6.62114  19.02705   2.54000
   2   1   45  101   0.19749   0.33516   0.09666  14.24408  19.01995   1.06680
   2   2   44  141   0.10573   0.15612   0.09712   4.25324  18.72383   1.06680
   3   3
       3         1         2   3.86080   0.08413   2.23290
   0   0    0    0    0
   1   1   55   75   0.35076   0.13385   0.06791   6.25676   2.23290   2.54000
   1   2   78  159   0.22860   0.05993   0.07320   6.43145   1.65354   2.54000
   2   1   78  160   0.04065   0.03088   0.11286   2.00198   1.65225   1.32080
   2   2   80  210   0.32064   0.21715   0.11863   3.88022   1.29225   1.32080
   3   0   80  210   1.29225   4.80906   0.09608   6.21113   1.29225   0.00000
       3         2         2   0.45720   0.11688   1.81612
   0   0    0    0    0
   1   1   52   75   0.25695   0.07937   0.08759   2.95390   1.81612   0.22860
   1   2   57   82   0.17789   0.02540   0.13229   1.26506   1.38129   0.22860
   2   1   59   82   0.06104   0.23495   0.08822   3.13323   1.37742   0.22860
   2   2   61   89   0.02864   0.09207   0.05946   1.86921   1.29161   0.22860
   3   0   61   89   1.29161   2.73707   0.09285   8.47958   1.29161   0.00000
       3         3         2   1.90500   0.13368   4.60644
   2   1    2    0    0
   1   1   53    1   0.10792   0.10628   0.30019   8.08631   4.60192   0.96520
   1   2   54   38   0.12692   0.09959   0.33824  11.72690   4.36709   0.96520
   2   1   54   39   0.08316   0.08718   0.14369   6.17837   4.36386   0.93980
   2   2   52   75   0.23232   0.21899   0.10577   4.21625   4.05160   0.93980
   4   3
       4         1         7  17.75459   0.08203   9.03740
   0   0    0    0    0
   1   1   97   61   0.92169   0.21891   0.08429   5.77683   9.03740   2.54000
   1   2  103  155   0.86927   0.18199   0.07675   5.50497   7.24643   2.54000
   2   1  103  156   0.17884   0.23270   0.09639   5.82856   7.24514   2.54000
   2   2  103  254   0.07857   0.07697   0.10880   6.97524   6.98902   2.54000
   3   1  103  255   0.42680   0.19037   0.11561   5.30627   6.98773   2.54000
```

```
3   2  100  351   0.32223   0.11914   0.10468   5.04964   6.23999   2.54000
4   1  100  352   0.39860   0.12136   0.11925   4.50972   6.23869   2.54000
4   2   95  450   0.66656   0.17780   0.06861   5.08234   5.17483   2.54000
5   1   95  451   0.75928   0.28734   0.08877   4.19564   5.17418   2.54000
5   2  100  546   0.55555   0.18600   0.08876   4.01840   3.85999   2.54000
6   1  100  547   0.74316   0.23515   0.06018   6.52774   3.85935   2.54000
6   2   91  643   0.79619   0.22572   0.06993   6.83241   2.32064   2.54000
7   1   91  644   0.37271   0.21720   0.10283   8.10277   2.31548   2.51460
7   2   92  741   0.63180   0.34031   0.09229   5.12645   1.31613   2.51460
8   0   92  741   1.31613   2.92002   0.10564  10.18845   1.31613   0.00000
    4         2         2   0.63500   0.11270   1.58000
0   0    0    0   0
1   1   96   61   0.07094   0.05080   0.09996   2.10860   1.58000   0.33020
1   2   87   72   0.19099   0.11642   0.09164   1.85969   1.31806   0.33020
2   1   86   73   0.00441   0.04387   0.05690   2.02292   1.31484   0.30480
2   2   80   83   0.02204   0.20782   0.09170   2.51998   1.29161   0.30480
3   0   80   83   1.29161   4.69900   0.10575   9.71462   1.29161   0.00000
    4         3         2   1.54940   0.13546  11.21159
2   1    2    0   0
1   1   90    1   0.27054   0.17862   0.18024  10.85843  11.20642   0.78740
1   2   94   31   0.11914   0.05326   0.26167   9.65033  10.81675   0.78740
2   1   95   32   0.12222   0.30057   0.10319   5.72830  10.81546   0.76200
2   2   96   61   0.06616   0.13801   0.13610   2.12109  10.62836   0.76200
5   3
    5         1         3   7.59460   0.07485   3.99483
0   0    0    0   0
1   1  117  442   0.33930   0.12905   0.09157   3.97465   3.99483   2.54000
1   2  107  540   0.28844   0.08415   0.09596   4.66692   3.36709   2.54000
2   1  107  541   0.13615   0.05159   0.10862   6.14227   3.36580   2.54000
2   2  107  637   0.62062   0.21106   0.07925   8.63462   2.61032   2.54000
3   1  107  638   0.17737   0.04310   0.10307   4.38949   2.60580   2.51460
3   2  107  736   1.12650   0.24836   0.08731   6.52594   1.30645   2.51460
4   0  107  736   1.30645   2.86845   0.10661  10.13435   1.30645   0.00000
    5         2         2   2.64160   0.07788   1.97741
0   0    0    0   0
1   1  118  442   0.26572   0.09767   0.15984   5.37672   1.97741   2.54000
1   2  121  538   0.19363   0.04792   0.08159   4.88031   1.51806   2.54000
2   1  121  539   0.00881   0.17780   0.14701   4.36518   1.51742   0.10160
2   2  122  542   0.00409   0.05715   0.07840   1.94963   1.50516   0.10160
3   0  122  542   1.50516   3.14498   0.07477   7.12425   1.50516   0.00000
    5         3         5  11.65860   0.08735  11.45159
2   1    2    0   0
1   1  113    1   1.06648   0.14009   0.08207  11.12572  11.44965   2.54000
1   2  122   97   0.53609   0.04713   0.20265  15.86980   9.84707   2.54000
2   1  122   98   0.35041   0.29043   0.07969   6.45339   9.84514   2.54000
2   2  126  196   0.10895   0.06491   0.10982   4.71587   9.38772   2.54000
3   1  127  197   0.97814   0.29577   0.10571   6.41231   9.38578   2.54000
3   2  109  286   0.85669   0.24130   0.13449  12.13389   7.55289   2.54000
4   1  109  287   0.26340   0.15162   0.13976   6.57118   7.55159   2.54000
4   2  114  384   0.75918   0.41703   0.10271  11.44687   6.53031   2.54000
5   1  114  385   0.24733   0.28816   0.09065   4.58663   6.47418   1.49860
5   2  117  442   0.30750   0.34421   0.08958   5.52319   5.97547   1.49860
6   1
```

```
    6         1        2  4.95300  0.08368   3.32257
0    0    0    0    0
1    1  124    1  0.71457  0.17885  0.12188  20.34286   3.31160   2.54000
1    2  130   97  0.72284  0.15397  0.14767  13.85037   1.87419   2.54000
2    1  130   98  0.16660  0.10215  0.08571   7.24086   1.87354   2.41300
2    2  136  190  0.41276  0.22587  0.08178   6.70572   1.29484   2.41300
3    0  136  190  1.29484  4.92601  0.11446  18.92900   1.29484   0.00000
7    1
    7         1        2  3.55600  0.09438   2.39032
0    0    0    0    0
1    1  150    1  0.23242  0.07040  0.12794  16.65839   2.38967   2.54000
1    2  157   92  0.66564  0.19160  0.19299  24.00627   1.49161   2.54000
2    1  158   92  0.14533  0.24397  0.10461   7.01147   1.48709   1.01600
2    2  154  129  0.05596  0.07085  0.07655   3.92098   1.29032   1.01600
3    0  154  129  1.29032  3.54676  0.08603  26.67866   1.29032   0.00000
8   15
    8         1        2  4.11480  0.07683   2.67096
0    0    0    0    0
1    1  150  563  0.91928  0.21953  0.07957   6.83688   2.67096   2.54000
1    2  153  660  0.26975  0.03991  0.07377   6.57318   1.48193   2.54000
2    1  152  661  0.14489  0.28523  0.10510   4.34514   1.44967   1.57480
2    2  148  721  0.04607  0.06412  0.10948   3.93251   1.29096   1.57480
3    0  148  721  1.29096  3.81000  0.09605   9.25674   1.29096   0.00000
    8         2        2  4.41960  0.07727   2.57225
0    0    0    0    0
1    1  149  563  0.34566  0.13666  0.07314   6.96151   2.57225   2.54000
1    2  138  654  0.24015  0.07592  0.09545   6.51610   1.98645   2.54000
2    1  137  654  0.23200  0.11482  0.09493   4.00523   1.98387   1.87960
2    2  127  726  0.40219  0.17571  0.10891   4.01965   1.35225   1.87960
3    0  127  726  1.35225  3.30442  0.10148   9.72166   1.35225   0.00000
    8         3        2  5.00380  0.07499   2.60709
0    0    0    0    0
1    1  163  542  0.25880  0.08406  0.07763   5.62918   2.60709   2.54000
1    2  160  638  0.21281  0.04661  0.09190   2.97051   2.13548   2.54000
2    1  159  639  0.30718  0.13705  0.09292   4.73607   2.13483   2.46380
2    2  159  734  0.33346  0.12568  0.09433   4.72626   1.49484   2.46380
3    0  159  734  1.49484  3.36434  0.10246   9.32231   1.49484   0.00000
    8         4        2  1.52400  0.06947   1.76709
0    0    0    0    0
1    1  164  542  0.11235  0.06773  0.08079   2.08461   1.76709   0.76200
1    2  163  571  0.11797  0.04657  0.08403   1.98476   1.53677   0.76200
2    1  163  572  0.01901  0.03481  0.07092   1.76864   1.53225   0.76200
2    2  163  598  0.14035  0.22672  0.10056   2.43513   1.37742   0.76200
3    0  163  598  1.37742  5.06476  0.09538  12.69095   1.37742   0.00000
    8         5        2  1.27000  0.08681   1.74709
0    0    0    0    0
1    1  146  393  0.06588  0.10933  0.14104  16.76308   1.74709   0.63500
1    2  139  415  0.03477  0.03313  0.35409  19.23536   1.64645   0.63500
2    1  139  416  0.23200  0.28956  0.11422   6.68699   1.64516   0.63500
2    2  136  440  0.10865  0.10973  0.29469   8.81393   1.30580   0.63500
3    0  136  440  1.30580  2.52095  0.08693  12.41614   1.30580   0.00000
    8         6        2  0.38100  0.04963   1.42064
0    0    0    0    0
```

```
1    1  146  393   0.00387   0.01905   0.17439   1.49443   1.42064   0.20320
1    2  148  400   0.01548   0.05397   0.42584   6.97205   1.40129   0.20320
2    1  148  401   0.00719   0.04717   0.30600   1.33095   1.39355   0.17780
2    2  149  407   0.02378   0.13063   0.58320   7.79215   1.37032   0.17780
3    0  149  407   1.37032   2.59182   0.09313  16.31160   1.37032   0.00000
     8         7         4   9.52500   0.08236   3.23483
0    0    0    0   0
1    1  161  251   0.32837   0.11642   0.18106  10.90469   3.23483   2.54000
1    2  178  346   0.36195   0.09948   0.08853  10.34006   2.54451   2.54000
2    1  177  347   0.16729   0.08357   0.09496  14.98688   2.52967   2.54000
2    2  178  446   0.20949   0.07925   0.12853  13.75768   2.16774   2.54000
3    1  178  447   0.28290   0.08346   0.22050   4.67181   2.16645   2.54000
3    2  177  544   0.30807   0.06480   0.07537   5.17213   1.57677   2.54000
4    1  176  545   0.15477   0.12598   0.08471   4.31346   1.57483   1.90500
4    2  180  619   0.12523   0.07654   0.07784   4.78613   1.29677   1.90500
5    0  180  619   1.29677   6.39572   0.09159  10.52803   1.29677   0.00000
     8         8         2   0.27940   0.02971   1.36580
0    0    0    0   0
1    1  172   94   0.00290   0.00847   0.09159   2.00000   1.36580   0.15240
1    2  172   99   0.04355   0.10160   0.17013   2.80818   1.31935   0.15240
2    1  172  100   0.00166   0.03048   0.09748   2.15252   1.31806   0.12700
2    2  172  104   0.01382   0.22860   0.20066   2.18097   1.30387   0.12700
3    0  172  104   1.30387   5.31837   0.09469   7.82783   1.30387   0.00000
     8         9         2   2.05740   0.07605   5.87741
2    1    2    0   0
1    1  162  484   0.12142   0.08318   0.09855   2.90413   5.87741   1.04140
1    2  153  523   0.21341   0.12001   0.08486   4.03973   5.54257   1.04140
2    1  152  524   0.12262   0.16573   0.09688   3.24360   5.53741   1.01600
2    2  150  563   0.16318   0.19240   0.05956   5.43725   5.25676   1.01600
     8        10         2   1.52400   0.07573   4.78902
2    3    4    0   0
1    1  162  484   0.05294   0.05781   0.13149   2.07828   4.78902   0.76200
1    2  164  512   0.06384   0.04554   0.10581   2.31056   4.67225   0.76200
2    1  164  513   0.05996   0.06604   0.14892   2.84262   4.66902   0.76200
2    2  164  542   0.23359   0.22945   0.08192   4.39897   4.37870   0.76200
     8        11         2   3.37820   0.07498  11.88513
2    9   10    0   0
1    1  157  358   0.10931   0.04091   0.24246   8.21503  11.88513   2.54000
1    2  164  452   0.70681   0.24090   0.11312   6.49104  11.06901   2.54000
2    1  164  453   0.08430   0.07541   0.12260   2.46433  11.06513   0.83820
2    2  162  484   0.31506   0.26114   0.12946   6.00386  10.66965   0.83820
     8        12         2   1.06680   0.04837   3.88580
2    5    6    0   0
1    1  157  358   0.29864   0.18627   0.29761  21.08308   3.88580   0.53340
1    2  147  372   0.29619   0.18119   0.18601   5.55282   3.29096   0.53340
2    1  147  373   0.02229   0.11974   0.80775  20.08529   3.28580   0.53340
2    2  146  393   0.08352   0.42454   0.24428  11.65843   3.18515   0.53340
     8        13         2   2.87020   0.09327  16.59932
2   11   12    0   0
1    1  160  251   0.31954   0.12887   0.10781  27.36479  16.59932   2.54000
1    2  157  345   0.40304   0.13475   0.17384  15.23847  15.87674   2.54000
2    1  157  346   0.01548   0.04689   0.21044   1.61524  15.87351   0.33020
2    2  157  358   0.08194   0.22665   0.20941   3.14380  15.77932   0.33020
```

```
         8        14         2   4.19100   0.08542  21.90059
    2   13    7    0    0
    1    1  172   94   0.37059   0.10107   0.15437   4.99064  21.90059   2.54000
    1    2  156  188   1.18683   0.30560   0.13485   8.53601  20.34318   2.54000
    2    1  156  189   0.15306   0.15966   0.15987   6.99313  20.33802   1.65100
    2    2  160  251   0.34371   0.33060   0.16256   6.01872  19.84640   1.65100
         8        15         2   2.48920   0.09993  23.67672
    2   14    8    0    0
    1    1  179    1   0.07517   0.06220   1.77237  24.29467  23.67220   1.24460
    1    2  179   49   0.07580   0.03732   0.08637   6.38077  23.52123   1.24460
    2    1  179   50   0.09722   0.08636   0.53326  12.88070  23.51930   1.24460
    2    2  172   94   0.10407   0.06604   0.13804   3.02680  23.31994   1.24460
    9    1
         9         1         0   0.00000   0.00000   0.00000
    0    0    0    0    0
    0    0    0    0   4.98708   1.37065   0.09492  23.64940   4.98708   0.00000
```