DEVELOPMENT OF A REVERSE FLOOD ROUTING TECHNIQUE USING THE IMPLICIT METHOD

by

Robert N. Eli, II

Thesis submitted to the Graduate Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

in

Civil Engineering

APPROVED:

J. M. Wiggert, Chairman

D. N. Contractor

V. S. Rao

August, 1972

Blacksburg, Virginia

ACKNOWLEDGEMENTS

The writer wishes to express gratitude to Dr. J. M. Wiggert, Committee Chairman, Dr. D. N. Contractor, and Dr. V. S. Rao for their enthusiastic guidance which has made the past year an enjoyable and educational experience.

Additional credit should also be given to Dr. J. M. Wiggert and Dr. D. N. Contractor for the origination of the idea which has become the subject of this paper.

TABLE OF CONTENTS

		Page
I.	INTRODUCTION	1
II.	LITERATURE REVIEW	3
III.	BASIC THEORY	10
IV.	DEVELOPMENT OF THE COMPUTER MODEL	15
V.	DISCUSSION OF THE RESULTS	32
VI.	CONCLUSIONS AND RECOMMENDATIONS	45
VII.	REFERENCES	47
APPENDIX A		48
APPENDIX B		67
VITA		85

LIST OF TABLES

Table No. Page

1. River Station Features from Reusen's Dam to Bent Creek....................................34

LIST OF FIGURES

Figure No.		Page
1.	The x-t Plane Used in the Implicit Method	8
2.	Definition Sketch for the Development of the Continuity Equation	11
3.	Definition Sketch for the Development of the Momentum Equation	13
4.	Grid Subscripting Scheme	16
5.	Method of Approximation of Station Cross Sections	35
6.	Actual Discharge Hydrograph at Reusen's Dam	36
7.	Computed Discharge Hydrograph and Actual Discharge Hydrograph at Bent Creek	37
8.	Replotted Portion of Figure 7 on Expanded Time Scale	38
9.	Computed Discharge Hydrograph and Actual Discharge Hydrograph at Reusen's Dam	40
10.	Replotted Portion of Figure 9 on Expanded Time Scale	41
11.	Actual Discharge Hydrograph at Bent Creek	42
12.	Hypothetical Discharge Hydrograph at Bent Creek and Resulting Computed Discharge Hydrograph at Reusen's Dam	44

I. INTRODUCTION

In recent years numerous papers have been published on the subject of numerical solution of the equations of conservation of mass and momentum, as written for unsteady flow in open channels. These basic equations have been applied to a wide range of problems including surges in canals, tidal motions, reservoir regulation, and flood routing.

In particular, the regulation of reservoir discharges, whether by scheduled releases or as the result of fluctuating power demand on hydroelectric facilities, is of interest due to the wide ranging effects this unsteady flow situation can have at points downstream.

Application of the equations of unsteady flow to fluctuating reservoir releases has been successfully accomplished and is documented in the technical literature (references 4 and 5). The procedure consists of integrating the one-dimensional partial differential equations of continuity and momentum, using one of the recognized techniques of numerical integration. At the upstream end of the river reach a discharge (or stage) hydrograph can be specified and routed through the reach, yielding as a solution, the hydrograph at the lower end of the reach.

Of potential usefulness would be the ability to accomplish the reverse, that is, route a hydrograph from the lower end of the reach in the upstream direction, yielding as a solution the hydrograph necessarv at the upper end of the reach. As an example, a water pollution problem at some point downstream from a reservoir might require that water be released from the impoundment in order to provide a diluting effect. To conserve water it would be desirable to know the discharge versus time relationship required at the reservoir to provide the minimum needed diluting effect. If a reverse flood routing technique were available, the desired reservoir release schedule could be determined based on the discharge versus time relationship required at the downstream point.

It is the purpose of this paper to develop a reverse flood routing technique. The partial differential equations of flow are integrated numerically using the implicit method of expressing the governing equations in finite difference form. The implicit method was chosen due to the relative freedom permitted in the choise of time interval Δt and distance interval Δx. Other advantages include good stability characteristics and computational efficiency.

The resulting computer model was tested utilizing a short reach of the James River, Virginia. The section began at Reusen's Dam and extended 37 miles downstream to a gaging station at Bent Creek. During the time period chosen, most of the flow resulted from turbine output at the dam which was highly varied depending on the electrical demand. This resulted in a jagged, swiftly changing discharge hydrograph which provided a good test of computer model stability.

II. LITERATURE REVIEW

In 1957, J. J. Stoker (8) outlined the procedure of applying the method of characteristics to obtain the solution of the differential equations describing the propagation of surface waves in shallow water. He noted that only in the most simple cases would the characteristic method lend itself to explicit solutions. For the more general case the method of successive approximation was first presented as a method of determining the characteristics and thus the solution. The continuous region where the solution was desired, the x-t plane, was replaced by a series of points joined by short line segments called a characteristic net. The short line segments approximated the characteristic lines and the solution proceeded along the segments to each net point in the direction of positive time.

A disadvantage of the characteristic method, recognized by Stoker, was that the location of the net points could only be determined in course of the computation. Thus, it was necessary to compute the coordinates of the net points in addition to the values of the unknowns. In order to avoid this problem, Stoker proposed a fixed grid system for use in calculating the solutions numerically. The grid consisted of a fixed rectangular net in the x-t plane. The governing differential equations of flow were approximated by difference equations. The solution progressed in the direction of positive time with the unknowns along each row of points being explicitly determined from the preceding row of points by the solution of the difference equations. Although no direct use of the characteristic equations was necessary in arriving at the solutions, the time interval Δt had to be small enough to remain in the "domain of dependence" as determined by the characteristic lines and the distance interval Δt. The details of the characteristic and explicit methods of solution are thoroughly covered in reference (8).

In 1968, Baltzer and Lai (2) investigated three techniques for simulating transient flows using the digital computer to numerically solve the equations of continuity and momentum.

The first method of solution studied was the power series method utilizing a Maclaurin series expansion of the governing partial differential equations. The expansion was made for a given point in time at a given point along the reach in terms of water stage. In this manner the stage at a nearby point, Δx away, could be determined.

The second method investigated was solution by the method of characteristics. Unlike Stoker (8), the authors were able to use a fixed rectangular grid of points instead of having to determine the location of the points during the computational procedure. A linear interpolation scheme allowed the characteristic lines to be determined such that they passed directly through the points at which the unknowns were being computed. It was still necessary that the choice of time interval Δt be directly influenced by the characteristic lines and the choice of distance interval Δx.

The third method studied was the implicit method. As with the modified characteristic method, the authors were able to use a fixed rectangular grid of points. The implicit method required that the difference equations be written in terms of each rectangle of four points, resulting in two equations containing four unknowns. In writing the equations for an entire row of N points plus two boundary condition relations, a system of 2N equations in 2N unknowns resulted. Simultaneous solution of the equations enabled the procedure to advance to the next row of points where the solution process was repeated. The implicit method permitted an efficient numerical solution since the grid points no longer had to be within the "domain of dependence" of the characteristic lines as required in explicit methods. Large time intervals Δt could be chosen without significant loss of accuracy.

In 1970, Theodor Strelkoff (9) discussed the solution of the Saint-Venant equations, a form of the one-dimensional partial differential equations of unsteady flow. The various finite difference schemes available for their solution were analyzed with special attention paid to their stability characteristics.

The author explained that finite difference techniques such as the explicit and implicit schemes could fall prey to instabilities resulting from error buildup. It was pointed out that the truncation error is automatically introduced as a part of any finite difference scheme, the more rapidly the functions change with respect to x and t, the greater the error. And further, the numerical solution of differential equations is often complicated by the fact that the small errors of truncation and round-off are amplified during successive time steps of the calculation. Eventually, the solution may be completely masked by these errors.

In examining the explicit method with rectangular grid presented by Stoker (8), Strelkoff indicated that the greatest accuracy is obtained when the time interval Δt is closest to its limiting value. This relationship was first reported by Courant in 1959, and is known as the Courant condition.

The implicit method was also examined by Strelkoff. He noted that the finite difference equation could be written in several different forms, not all of which would yield stable solutions. It was recognized that the implicit scheme would result in a large number of equations that would have to be solved simultaneously using matrix techniques. In order to reduce the computer storage required for the matrix solution of 2N equations for 2N unknowns, the author reported on the "double-sweep" method of solving the equations, which took advantage of the fact that most of the resulting matrix was occupied by zeroes. Details of the method can be obtained in reference (9).

Also in 1970, M. Amein and C. Fang (1) investigated the relative merits of the three most popular computation schemes for solving the partial differential equations of unsteady, open channel flow. Many computational runs were made utilizing the three methods; the explicit, the characteristic, and the implicit. Flood hydrographs were routed through actual river channels and rectangular channels, utilizing each of the three computational schemes for comparison purposes. The solution of the equation of unsteady flow by the three different methods was in reasonable agreement with the observed values. An analysis of the computer time required revealed the efficiency of the implicit method over the other methods. The authors found that large time intervals could be utilized in the implicit method with no significant loss of accuracy. The explicit method required small time intervals in comparison.

In applying the solution methods to a real channel with rapidly varying cross-sectional properties, the authors' attempts to find solutions using the characteristic and explicit methods were unsuccessful. The implicit method was found to be the only method that could handle significant changes in cross section from station to station without difficulty. The authors concluded that the implicit method demonstrated greater stability and was more efficient than the other two methods in the solution of the one-dimensional partial differential equations of unsteady flow as applied to natural channels.

In 1972, Contractor and Wiggert (4) applied the implicit method of solution to the one-dimensional, partial differential equations of unsteady flow on the James River, Virginia.

The authors established a fixed rectangular grid in the x-t plane as shown in Figure 1. The finite difference equations were first written for the point $\hat{R}$ by taking the appropriate property values at the points A, B, C, and D. The computer model written using these difference equations yielded solutions that tended to oscillate about the actual solution, resulting in a jagged appearance when plotted. This was determined to occur only when a slowly changing hydrograph was routed through a reach. As long as the rate of change of the hydrograph

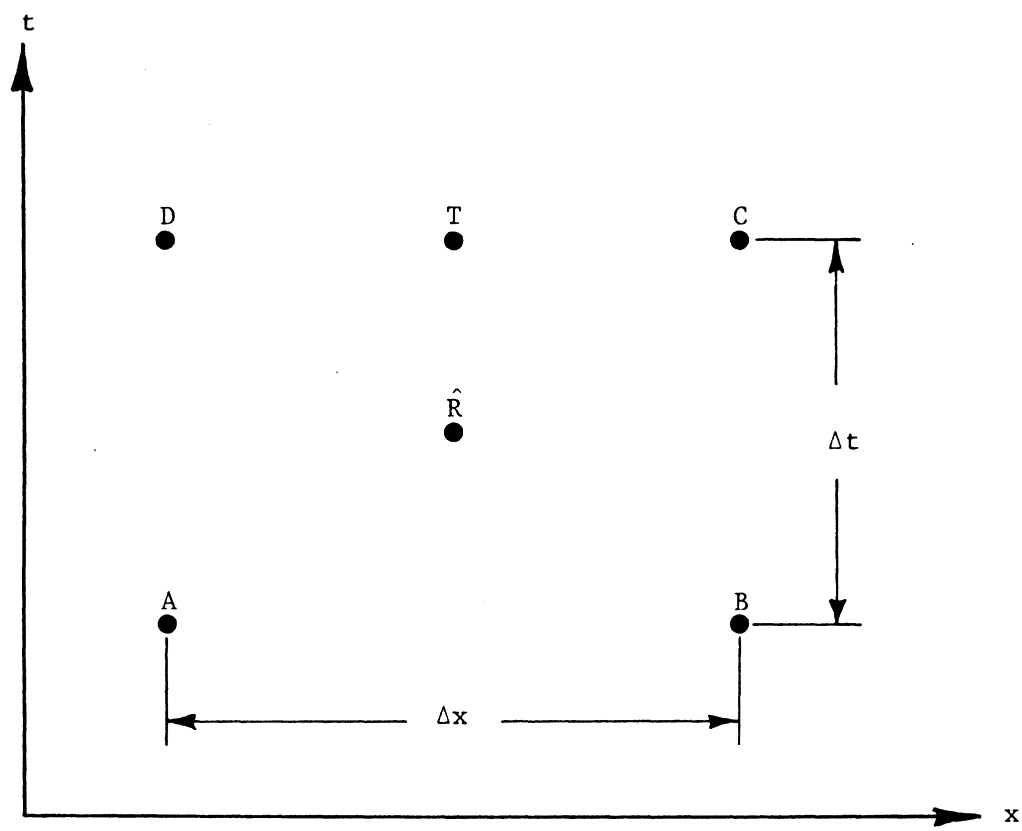

Figure 1. The x-t plane used in the implicit method (Contractor and Wiggert (4)).

was large in relation to the time interval Δt, the point $\hat{R}$ resulted in the greatest accuracy.

The finite difference equations were next written for the point T by again taking the differences (with appropriate consideration for the off-center location of T) at the points A, B, C, and D. The computer model written using these difference equations produced a smooth hydrograph that compared well with actual values in those cases where the hydrograph was not rapidly changing in relation to the time interval Δt. Using the point T for rapidly varying situations produced a smoothed hydrograph that did not accurately represent the actual data.

III. BASIC THEORY

Before proceeding with the details of numerical solution of the basic equations governing unsteady flow in open channels of varying cross section, it is appropriate at this point to develop these equations, continuity and momentum, from first principles. Of additional significance is the form of each of these equations which the writer will develop. In the literature (references 1, 4, and 5), researchers have invariably manipulated the two basic equations to a form containing average velocity in place of discharge. In doing so, it is almost always necessary to eliminate higher order terms in order to facilitate solution of the equations. The writer has left the equations in the more basic, and he believes simpler form, by not converting discharge to average velocity.

Development of the one-dimensional, unsteady, partial differential equation of continuity can be followed by referring to Figure 2.

The discharge changes with distance along the channel at a rate equal to:

$$\frac{\partial Q}{\partial x} - q \qquad [1]$$

where Q is the discharge in the channel and q the lateral inflow per unit length. The difference in the total volume through sections 1 and 2 during the time dt is:

$$\left(\frac{\partial Q}{\partial x}\right) dxdt - q\ dxdt \qquad [2]$$

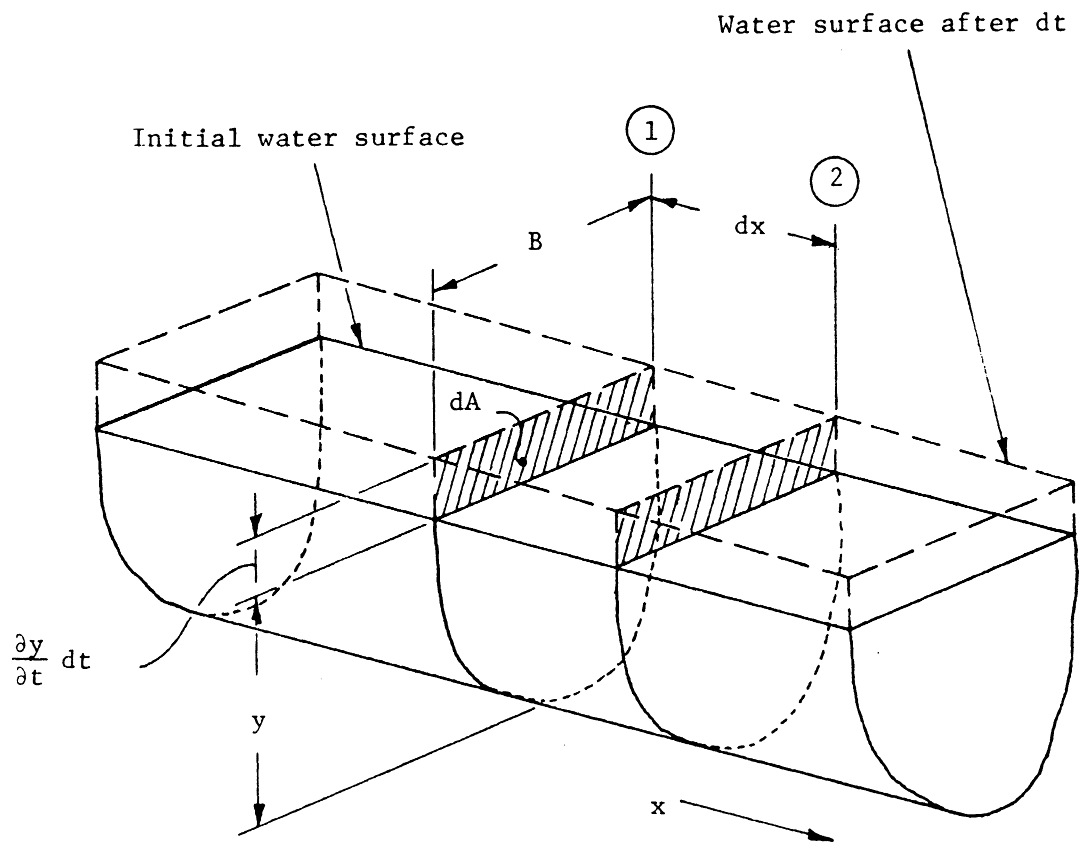

Figure 2. Definition sketch for the development of the continuity equation.

In addition the corresponding change in channel storage within the element during the time dt is:

$$B\left(\frac{\partial y}{\partial t}\right) dxdt \qquad [3]$$

The net change in total volume plus the change in storage should be zero; therefore:

$$\left(\frac{\partial Q}{\partial x}\right) dxdt - q\ dxdt + B\left(\frac{\partial y}{\partial t}\right) dxdt = 0 \qquad [4]$$

Simplifying equation [4], the final form becomes

$$\frac{\partial Q}{\partial x} + \frac{\partial A}{\partial t} - q = 0 \qquad [5]$$

The development of the one-dimensional unsteady, partial differential equation of motion follows the usual steps with reference being made to Figure 3. The sum of the forces on the fluid element must equal the rate of change in momentum. The net static force on the fluid element is equal to the rate of change of the static force over the distance dx:

$$\frac{\partial(\gamma\bar{y}A)}{\partial x}\ dx \qquad [6]$$

For gentle bed slopes the weight component acting downstream is given by:

$$\gamma S_o A dx \qquad [7]$$

where S_o is the bed slope. The shear force is:

$$-\gamma S_f A dx \qquad [8]$$

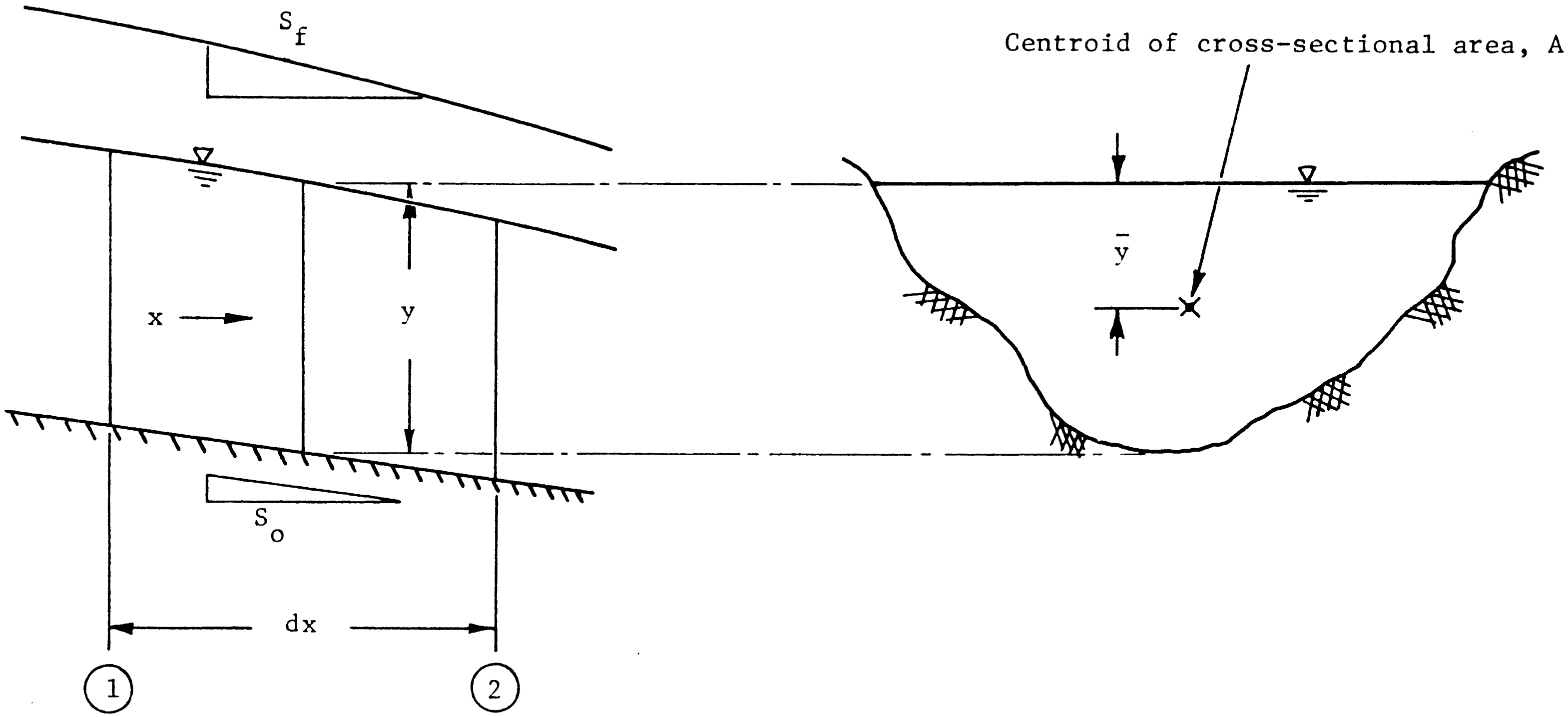

Figure 3. Definition sketch for the development of the momentum equation.

where S_f is the friction slope. The sum of the forces must equal the total rate of change in momentum:

$$d(\rho VQ) \qquad [9]$$

where V is the average velocity. Equating the sum of the force terms to the momemtum term results in the equation of motion:

$$\frac{\partial(\gamma\bar{y}A)}{\partial x}\,dx + \gamma S_o A dx - \gamma S_f A dx = d(\rho VQ) \qquad [10]$$

which can be simplified to:

$$\frac{\partial(\bar{y}A)}{\partial x}\,dx + (S_o - S_f)\,Adx = \frac{1}{g}\,d(\frac{Q^2}{A}). \qquad [11]$$

The continuity equation [5] and the equation of motion [11] together are sufficient to fully describe the one dimensional, unsteady flow in channels of varying cross section.

All that remains is to integrate these equations numerically over a given reach of river for a given transient flow condition by utilizing the high-speed digital computer.

IV. DEVELOPMENT OF THE COMPUTER MODEL

The one-dimensional, unsteady, partial differential equations of continuity and momentum have previously been developed and are restated here:

$$\frac{\partial Q}{\partial x} + \frac{\partial A}{\partial t} - q = 0 \qquad [12]$$

$$\frac{\partial (\bar{y}A)}{\partial x} \; dx + (S_o - S_f) \; Adx = \frac{1}{g} \; d \; (\frac{Q^2}{A}). \qquad [13]$$

In solving these equations numerically by using the implicit method, a fixed grid of points must be established in the x-t plane. The grid points are spaced in accordance with the desired time interval and distance divisions along the reach. A subscripting scheme was used in labeling the points as shown in Figure 4.

The first step in the numerical solution of equations [12] and [13] requires the replacement of the partial and total differentials with equivalent finite difference terms.

The writer made the decision to write the finite difference equations at the point $\hat{R}$ based on the findings of Contractor and Wiggert (4). In their research the point $\hat{R}$ gave the best results when computing rapidly varying flows.

In writing the finite difference form of the continuity equation, each term is replaced by an equivalent finite difference term written for the point $\hat{R}$ as follows:

$$\frac{\partial Q}{\partial x} = \left[\frac{(Q_{i+1,j} + Q_{i+1,j+1})}{2} - \frac{(Q_{i,j} + Q_{i,j+1})}{2} \right] \quad \frac{1}{\Delta x_i}$$

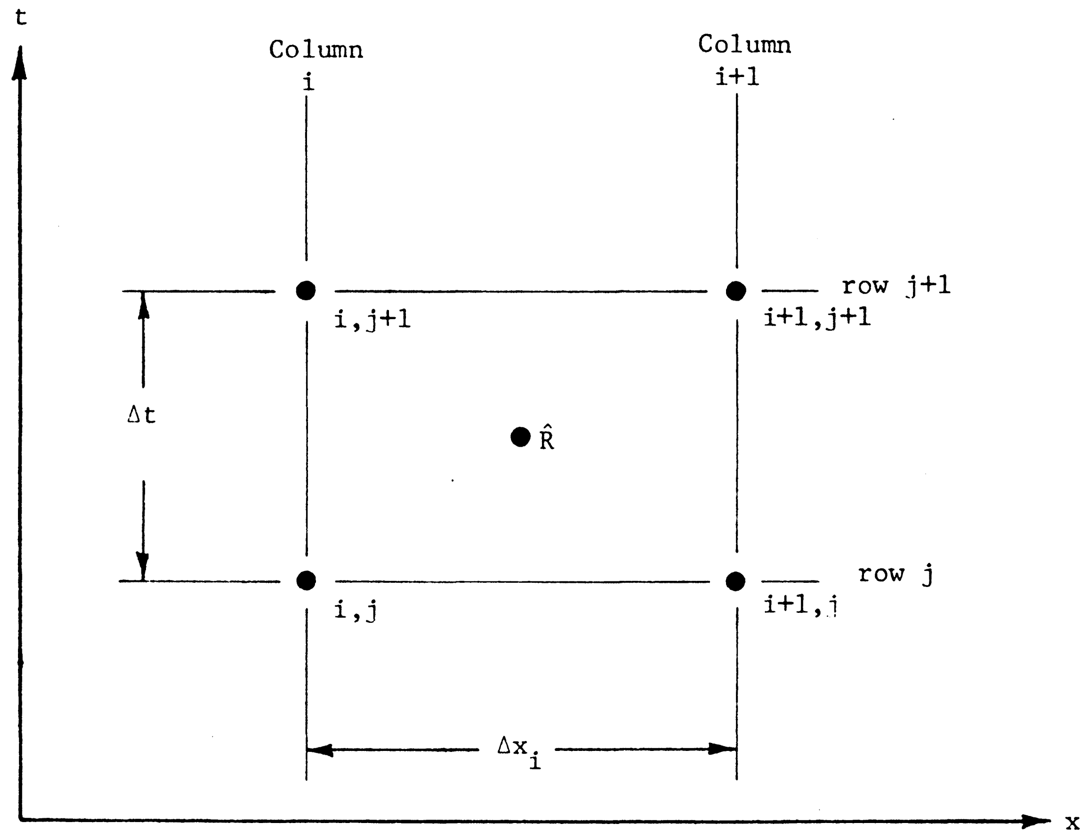

Figure 4. Grid subscripting scheme.

$$\frac{\partial A}{\partial t} = \left[\frac{(A_{i,j+1} + A_{i+1,j+1})}{2} - \frac{(A_{i,j} + A_{i+1,j})}{2} \right] \frac{1}{\Delta t}$$

$$- q = \text{constant} = - \text{QINFLO} \qquad [14]$$

where Δx_i can vary from one grid rectangle to the next. For convenience, Δt was chosen to be constant. The lateral inflow was assumed to be a known constant per unit length along the reach, and designated as "QINFLO."

The terms of the momentum equation are replaced by their equivalent finite-difference relations in a similar manner:

$$- \frac{\partial(\bar{y}A)}{\partial x} dx = \frac{\bar{y}A|_{i,j} + \bar{y}A|_{i,j+1}}{2} - \frac{\bar{y}A|_{i+1,j} + \bar{y}A|_{i+1,j+1}}{2}$$

$$(S_o - S_f)\, A dx = S_{o_i}\ \Delta x_i \left[\frac{A_{i,j} + A_{i+1,j} + A_{i,j+1} + A_{i+1,j+1}}{4} \right]$$

$$- \Delta x_i \left[\frac{S_f A|_{i,j} + S_f A|_{i+1,j} + S_f A|_{i,j+1} + S_f A|_{i+1,j+1}}{4} \right]$$

$$\frac{1}{g}\, d\left(\frac{Q^2}{A}\right) = \frac{1}{g} \left[\frac{Q^2}{A} \Big|_{i+1,j+1} - \frac{Q^2}{A} \Big|_{i,j} \right] \qquad [15]$$

where S_o, the bed slope, is a known value at each cross section of the river reach. The friction slope, S_f, is given by the Manning relation:

$$S_f = \frac{n^2 Q|Q|P^{4/3}}{2.2082\ A^{10/3}}$$

The only unknowns in the two finite equations are the dependent variables, discharge Q and depth y. The cross-section area A, and the distance from the water surface to the centroid $\bar{y}$, are both functions of depth y. Manning's roughness number, n, is also unknown, but a value is assumed initially and later adjusted to the value which results in a solution that best matches the actual data.

Up to this point the procedure is identical regardless of whether the governing equations are being solved in conjunction with forward flood routing or reverse flood routing. The writer first developed a computer model for forward flood routing in order to provide a basis for the development of the reverse flood routing solution technique. It was strongly suspected, and subsequently verified, that a large number of the solution steps would be common to both the forward and reverse flood routing techniques. For the above reason the procedure required for forward flood routing will be presented first.

The reach is first divided into the desired number of sections, N-1, of Δx_i length. Choosing a time interval Δt then completes a grid of points of N columns and a variable number of rows, M, depending on the time span of the upstream hydrograph being routed through the reach. In the implicit method, the j row values of Q and y are known and the j+1 row values unknown. Therefore, Q and y along the j=1 row must be assigned initial values in order to enable the solution to be advanced to the j=2 row. Since there are N grid columns, there are 2N unknowns. Two finite difference equations, continuity and momentum, can be written for each of N-1 grid rectangles. For each grid rectangle the equations are written in four unknowns. Since the unknowns at each grid point are common to the equations written for the two adjacent grid rectangles (with the exception of the upstream and downstream boundaries), a system of 2(N-1) equations in 2N unknowns results. The remaining two equations needed are provided by the boundary conditions. At the upstream boundary, i=1, the input discharge hydrograph is known, therefore.

$$Q_{1,j+1} - QU_{j+1} = 0 \qquad [17]$$

where QU_{j+1} is the discharge specified by the input discharge hydrograph for the j+1 time step. At the downstream boundary, i=N, the depth-discharge relationship is known, giving the last relationship needed:

$$y_{N,j+1} - yd = 0 \qquad [18]$$

where yd is the depth determined from the depth-discharge relationship for the computed value of discharge, $Q_{N,j+1}$. For convenience the complete set of 2N equations can be written as follows:

$$\begin{aligned}
&H_o(Q_1,y_1) = 0\\
&G_1(Q_1,y_1,Q_2,y_2) = 0\\
&F_1(Q_1,y_1,Q_2,y_2) = 0\\
&\vdots \qquad\qquad \vdots\\
&G_i(Q_i,y_i,Q_{i+1},y_{i+1}) = 0\\
&F_i(Q_i,y_i,Q_{i+1},y_{i+1}) = 0\\
&\vdots \qquad\qquad \vdots\\
&G_{N-1}(Q_{N-1},y_{N-1},Q_N,y_N) = 0\\
&F_{N-1}(Q_{N-1},y_{N-1},Q_N,y_N) = 0\\
&H_N(Q_N,y_N) = 0
\end{aligned} \qquad [19]$$

where H_o and H_N are the boundary equations, G_1 to G_{N-1} the continuity equations, and F_1 to F_{N-1} the momentum equations. The second subscript, j+1 is omitted to improve clarity.

Solution of the finite difference equations is accomplished by applying the generalized Newton iteration method to equation [19]. Trial values are assigned to the unknowns, which when substituted into equation [19] will in general result in non-zero values on the right hand sides of the equations called residues. The solution to the system of equations is arrived at by adjusting the trial values in successive iterative steps until the residues approach zero within a predetermined tolerance.

In the generalized Newton iteration method the residue and the partial derivatives of equation [19] are related as follows:

$$\frac{\partial H_o}{\partial Q_1}\, dQ_1 + \frac{\partial H_o}{\partial y_1}\, dy_1 = -R_1$$

$$\frac{\partial G_1}{\partial Q_1}\, dQ_1 + \frac{\partial G_1}{\partial y_1}\, dy_1 + \frac{\partial G_1}{\partial Q_2}\, dQ_2 + \frac{\partial G_1}{\partial y_2}\, dy_2 = -R_2$$

$$\frac{\partial F_1}{\partial Q_1}\, dQ_1 + \frac{\partial F_1}{\partial y_1}\, dy_1 + \frac{\partial F_1}{Q_2}\, dQ_2 + \frac{\partial F_1}{\partial y_2}\, dy_2 = -R_3$$

$$\vdots \qquad\qquad\qquad\qquad\qquad\qquad \vdots$$

$$\frac{\partial G_i}{\partial Q_i}\, dQ_i + \frac{\partial G_i}{\partial y_i}\, dy_i + \frac{\partial G_i}{\partial Q_{i+1}}\, dQ_{i+1} + \frac{\partial G_i}{\partial y_{i+1}}\, dy_{i+1} = -R_{2_i}$$

$$\frac{\partial F_i}{\partial Q_i}\, dQ_i + \frac{\partial F_i}{\partial y_i}\, dy_i + \frac{\partial F_i}{\partial Q_{i+1}}\, dQ_{i+1} + \frac{\partial F_i}{\partial y_{i+1}}\, dy_{i+1} = -R_{2_{i+1}}$$

$$\vdots \qquad\qquad\qquad\qquad\qquad\qquad\qquad \vdots$$

$$\frac{\partial G_{N-1}}{\partial Q_{N-1}} dQ_{N-1} + \frac{\partial G_{N-1}}{\partial y_{N-1}} dy_{N-1} + \frac{\partial G_{N-1}}{\partial Q_N} dQ_N + \frac{\partial G_{N-1}}{\partial y_N} dy_N = -R_{2(N-1)}$$

$$\frac{\partial F_{N-1}}{\partial Q_{N-1}} dQ_{N-1} + \frac{\partial F_{N-1}}{\partial y_{N-1}} dy_{N-1} + \frac{\partial F_{N-1}}{\partial Q_N} dQ_N + \frac{\partial F_{N-1}}{\partial y_N} dy_N = -R_{2(N-1)+1}$$

$$\frac{\partial H_N}{\partial Q_N} dQ_N + \frac{\partial H_N}{\partial y_N} dy_N = -R_{2N} \qquad [20]$$

where R denotes the residue.

Equation [20] forms a second system of 2N equations containing 2N unknowns in dQ and dy. The partial differential coefficients are known values which are evaluated from equation [19] as follows, noting that the j+1 subscript is again omitted for clarity:

$$\frac{\partial H_o}{\partial Q_1} = 1.$$

$$\frac{\partial H_o}{\partial y_1} = 0.$$

$$\frac{\partial G_i}{\partial Q_i} = -\frac{1}{2\Delta x_i}$$

$$\frac{\partial G_i}{\partial y_i} = \frac{1}{2\Delta t} \frac{\partial A_i}{\partial y_i}$$

$$\frac{\partial G_i}{\partial Q_{i+1}} = \frac{1}{2\Delta x_i}$$

$$\frac{\partial G_i}{\partial y_{i+1}} = \frac{1}{2\Delta t} \frac{\partial A_{i+1}}{\partial y_{i+1}}$$

$$\frac{\partial F_i}{\partial Q_i} = - \frac{\Delta x_i}{4} A_i \frac{\partial S_{f_i}}{\partial Q_i}$$

$$\frac{\partial F_i}{\partial y_i} = \frac{1}{2} \frac{\partial \bar{y}A|_i}{\partial y_i} + \frac{S_{o_i} \Delta x_i}{4} \frac{\partial A_i}{\partial y_i} - \frac{\Delta x_i}{4} S_{f_i} \frac{\partial A_i}{\partial y_i}$$

$$- \frac{\Delta x_i}{4} A_i \frac{\partial S_{f_i}}{\partial dy_i}$$

$$\frac{\partial F_i}{\partial Q_{i+1}} = - \frac{\Delta x_i}{4} A_{i+1} \frac{\partial S_{f_{i+1}}}{\partial Q_{i+1}} - \frac{2Q_{i+1}}{gA_{i+1}}$$

$$\frac{\partial F_i}{\partial y_{i+1}} = - \frac{1}{2} \frac{\partial \bar{y}A|_{i+1}}{\partial y_{i+1}} + \frac{S_{o_{i+1}}}{4} \frac{\partial A_{i+1}}{\partial y_{i+1}}$$

$$- \frac{\Delta x_i}{4} S_{f_{i+1}} \frac{\partial A_{i+1}}{\partial y_{i+1}} - \frac{\Delta x_i}{4} A_{i+1} \frac{\partial S_{f_{i+1}}}{\partial y_{i+1}}$$

$$+ \frac{1}{g} \frac{Q_{i+1}^2}{A_{i+1}^2} \frac{\partial A_{i+1}}{\partial y_{i+1}}$$

where

$$\frac{\partial A_i}{\partial y_i} = B_i$$

$$\frac{\partial A_{i+1}}{\partial y_{i+1}} = B_{i+1}$$

$$\frac{\partial S_{f_i}}{\partial Q_i} = \frac{n^2 \; Q_i \; P_i^{4/3}}{1.1041 \; A_i^{10/3}}$$

$$\frac{\partial \bar{y}A|_i}{\partial y_i} = A_i$$

$$\frac{\partial S_{f_i}}{\partial y_i} = - \frac{n^2 \; Q_i |Q_i| P_i^{4/3}}{0.66246 \; A_i^{13/3}} \; \frac{\partial A_i}{\partial y_i}$$

$$+ \frac{n^2 \; Q_i |Q_i| P_i^{1/3}}{1.65615 \; A_i^{10/3}} \; \frac{\partial P_i}{\partial y_i}$$

$$\frac{\partial S_{f_{i+1}}}{\partial Q_{i+1}} = \frac{n^2 \; Q_{i+1} \; P_{i+1}^{4/3}}{1.1041 \; A_{i+1}^{10/3}}$$

$$\frac{\partial \bar{y}A_{i+1}}{\partial y_{i+1}} = A_{i+1}$$

$$\frac{\partial S_{f_{i+1}}}{\partial y_{i+1}} = - \frac{n^2 Q_{i+1} |Q_{i+1}| P_{i+1}^{4/3}}{0.66246 \; A_{i+1}^{13/3}} \; \frac{\partial A_{i+1}}{\partial y_{i+1}}$$

$$+ \frac{n^2 Q_{i+1} |Q_{i+1}| P_{i+1}^{1/3}}{1.65615 \; A_{i+1}^{10/3}} \; \frac{\partial P_{i+1}}{\partial y_{i+1}}$$

$$\frac{\partial H_N}{\partial Q_N} = - \frac{\text{DIFFY}}{\text{DIFFQ}}$$

$$\frac{\partial H_N}{\partial y_N} = 1. \qquad [21]$$

It should be noted that DIFFY/DIFFQ is the slope of the downstream depth-discharge relationship (rating curve). The coefficients of the unknowns given by equation [21] form a banded matrix. The system of equations can be conveniently solved by using a program in the IBM Scientific Subroutine Package.

The solution of equation [20] yields N values of dQ and N values of dy which are the changes to be applied to the approximate values of Q and y, respectively, in order to make the approximate values approach the actual solution. Q + dQ and y + dy are then substituted back into equation [19] in order to produce a new series of residues which will then repeat the cycle. Iteration continues through as many cycles as are required for dQ and dy to become small enough values to pass the interval tolerance test. The values of Q and y at this point become the solutions to equation [19] at the j+1 step and the procedure advances to the j+2 step. The iterative process is then repeated for the j+2 step. Thus, in this manner the values of Q and y are determined along each row of grid points in turn, moving in the direction of positive time, until the Q and y values at all of the grid points in the x-t plane have been determined. This completes the computational procedure with the downstream discharge hydrograph being produced by simply plotting the Q values along column i=N as a function of time.

In attempting to accomplish reverse flood routing, two different variations in what was essentially the same solution procedure used in the forward flood routing technique were attempted. The first variation attempted was to write the difference equations in the same manner as was done in forward flood routing, with appropriate changes in the boundary condition equations. The j row of grid points became the unknown points and the j+1 row the known points. Therefore, the solution proceeded in the downward direction (in the direction of negative time) starting with the j=M-1 row at the top of the grid.

The j=M row, where M-1 is the maximum number of time intervals Δt, was initialized with values of Q and y in the same manner as the j=1 row was initialized in the forward flood routing technique. As before, a system of 2N equations with 2N unknowns results. The writer was unable to obtain convergence to a solution in the iterative procedure utilizing this method, and therefore attempted the second variation in solution procedure.

The second variation was unorthodox in that the solution proceeded in the negative x direction in the x-t plane. Unlike the other procedures presented, the solution is obtained column by column instead of row by row in the grid of points. To accomplish reverse flood routing utilizing this technique, the i=N column grid points are initialized with values of Q and y, the values of Q being taken from the downstream discharge hydrograph at the appropriate time increment. Corresponding values of depth, y, are obtained from a rating curve, giving the relationship between depth and discharge at the downstream boundary. This solution procedure allowed a solution to be obtained for all of the grid points and therefore will be presented in full detail below.

Since the basic finite difference equations (equations [14] and [15]) remain unchanged, the same grid of points used in the forward flood routing scheme can be used for the column by column reverse flood routing scheme. From this point on, the column by column solution technique will be referred to as the "column procedure" and the row by row technique as the "row procedure."

As before, we again have a grid of points of N columns by M rows.

This time, however, a slightly different criterion is required in the choice M. The total time period used must be sufficiently large to include the downstream discharge hydrograph plus a period of steady flow added to the beginning and end of the downstream hydrograph. This becomes necessary since a boundary value, either discharge or depth, must be known for all j=1 and j=M grid points. A sufficiently long steady flow period must be added to the discharge hydrograph to allow the unsteady flow solution to progress across the x-t plane without reaching the steady flow boundary values. Although no direct use of characteristic lines is made in the implicit solution technique, their effect must still be taken into account. Another way of stating the above restriction is that the characteristic lines along which the unsteady flow disturbances travel in the x-t plane must not intercept the upper or lower boundaries of the grid when using the column procedure of solution.

In computing the solution in the negative x direction using the column procedure, the difference equations of continuity and momentum are written for the grid rectangles in the vertical direction. This results in 2(M-1) equations containing 2M unknowns. The knowns are along the i+1 column and the unknowns along the i column.

Once again two additional equations are required to provide a full set of equations. These are provided by boundary values along row j=1 and j=M. Discharge, Q, is known along the rows since the regions of steady flow previously described are added on either end of the downstream discharge hydrograph. Therefore, the lower boundary equation is given by:

$$Q_{i,1} - (Q_{i+1,1} - DQDX) = 0 \qquad [22]$$

and the upper boundary condition given by:

$$Q_{i,M} - (Q_{i+1,M} - DQDX) = 0 \qquad [23]$$

where DQDX is the total lateral inflow occurring over the distance increment Δx_i. In the same manner as before, the complete set of equations can be written conveniently as:

$$\begin{aligned}
&H_o(y_1,Q_1) = 0\\
&G_1(y_1,Q_1,y_2,Q_2) = 0\\
&F_1(y_1,Q_1,y_2,Q_2) = 0\\
&\vdots \qquad\qquad \vdots\\
&G_j(y_j,Q_j,y_{j+1},Q_{j+1}) = 0\\
&F_j(y_j,Q_j,y_{j+1},Q_{j+1}) = 0\\
&\vdots \qquad\qquad \vdots\\
&G_{M-1}(y_{M-1},Q_{M-1},y_M,Q_M) = 0\\
&F_{M-1}(y_{M-1},Q_{M-1},y_M,Q_M) = 0\\
&H_M(y_M,Q_M) = 0
\end{aligned} \qquad [24]$$

where H_o and H_M are the boundary equations, G_1 to G_{M-1} the continuity equations, and F_1 to F_{M-1} the momentum equations. The first subscript, i, was omitted to improve clarity.

Solution of the finite difference equations is obtained by applying the generalized Newton iteration method to equation [24] in exactly the same manner as done in obtaining the forward flood routing solution. Residues resulting from the insertion into equation [24] of trial values of Q and y are related to the partial derivatives of equation [24] as follows:

$$\frac{\partial H_o}{\partial y_1} dy_1 + \frac{\partial H_o}{\partial Q_1} dQ_1 = -R_1$$

$$\frac{\partial G_1}{\partial y_1} dy_1 + \frac{\partial G_1}{\partial Q_1} dQ_1 + \frac{\partial G_1}{\partial y_2} dy_2 + \frac{\partial G_1}{\partial Q_2} dQ_2 = -R_2$$

$$\frac{\partial F_1}{\partial y_1} dy_1 + \frac{\partial F_1}{\partial Q_1} dQ_1 + \frac{\partial F_1}{\partial y_2} dy_2 + \frac{\partial F_1}{\partial Q_2} dQ_2 = -R_3$$

$$\vdots \qquad\qquad\qquad\qquad \vdots$$

$$\frac{\partial G_j}{\partial y_j} dy_j + \frac{\partial G_j}{\partial Q_j} dQ_j + \frac{\partial G_j}{\partial y_{j+1}} dy_{j+1} + \frac{\partial G_j}{\partial Q_{j+1}} dQ_{j+1} = -R_{2j}$$

$$\frac{\partial F_j}{\partial y_j} dy_j + \frac{\partial F_j}{\partial Q_j} dQ_j + \frac{\partial F_j}{\partial y_{j+1}} dy_{j+1} + \frac{\partial F_j}{\partial Q_{j+1}} dQ_{j+1} = -R_{2j+1}$$

$$\vdots \qquad\qquad\qquad\qquad \vdots$$

$$\frac{\partial G_{M-1}}{\partial y_{M-1}} dy_{M-1} + \frac{\partial G_{M-1}}{\partial Q_{M-1}} dQ_{M-1} + \frac{\partial G_{M-1}}{\partial y_M} dy_M + \frac{\partial G_{M-1}}{\partial Q_M} dQ_M = -R_{2(M-1)}$$

$$\frac{\partial F_{M-1}}{\partial y_{M-1}} dy_{M-1} + \frac{\partial F_{M-1}}{\partial Q_{M-1}} dQ_{M-1} + \frac{\partial F_{M-1}}{\partial y_M} dy_M + \frac{\partial F_{M-1}}{\partial Q_M} dQ_M = -R_{2(M-1)+1}$$

$$\frac{\partial H_M}{\partial y_M} dy_M + \frac{\partial H_M}{\partial Q_M} dQ_M = -R_{2M} \qquad [25]$$

Equation [25] forms a system of 2M equations containing 2M unknowns in dy and dQ. The partial differential coefficients are known values which are evaluated from equation [24] as follows, noting that the i subscript is being omitted for simplicity:

$$\frac{\partial H_o}{\partial y_1} = 0.$$

$$\frac{\partial H_o}{\partial Q_1} = 1.$$

$$\frac{\partial G_j}{\partial y_j} = - \frac{1}{2\Delta t} \frac{\partial A_j}{\partial Y_j}$$

$$\frac{\partial G_j}{\partial Q_j} = - \frac{1}{2\Delta x}$$

$$\frac{\partial G_j}{\partial y_{j+1}} = \frac{1}{2\Delta t} \frac{\partial A_{j+1}}{\partial y_{j+1}}$$

$$\frac{\partial G_j}{\partial Q_{j+1}} = - \frac{1}{2\Delta x}$$

$$\frac{\partial F_j}{\partial y_j} = \frac{1}{2} \frac{\partial \bar{y}A|_j}{\partial y_j} + \frac{S_o \Delta x}{4} \frac{\partial A_j}{\partial y_j} - \frac{\Delta x \; S_{f_j}}{4} \frac{\partial A_j}{\partial y_j} - \frac{\Delta x \; A_j}{4} \frac{\partial S_{f_j}}{\partial y_j}$$

$$- \frac{Q_j^2}{gA_j^2} \frac{\partial A_j}{\partial y_j}$$

$$\frac{\partial F_j}{\partial Q_j} = - \frac{\Delta x \; A_j}{4} \frac{\partial S_{f_j}}{\partial Q_j} + \frac{2Q_j}{gA_j}$$

$$\frac{\partial F_j}{\partial y_{j+1}} = \frac{1}{2} \frac{\partial \bar{y}A|_{j+1}}{\partial y_{j+1}} + \frac{S_o \Delta x}{4} \frac{\partial A_{j+1}}{\partial y_{j+1}} - \frac{\Delta x S_{f_{j+1}}}{4} \frac{\partial A_{j+1}}{\partial y_{j+1}}$$

$$- \frac{\Delta x A_{j+1}}{4} \frac{\partial S_{f_{j+1}}}{\partial y_{j+1}}$$

$$\frac{\partial F_j}{\partial Q_{j+1}} = - \frac{\Delta x \ A_{j+1}}{4} \frac{\partial S_{f_{j+1}}}{\partial Q_{j+1}}$$

where:

$$\frac{\partial A_j}{\partial y_j} = B_j$$

$$\frac{\partial A_{j+1}}{\partial y_{j+1}} = B_{j+1}$$

$$\frac{\partial \bar{y}A|_j}{\partial y_j} = A_j$$

$$\frac{\partial S_{f_j}}{\partial y_j} = - \frac{n^2 Q_j |Q_j| P_j^{4/3}}{0.66246 \ A_j^{13/3}} \frac{\partial A_j}{\partial y_j} + \frac{n^2 Q_j |Q_j| P_j^{1/3}}{1.65615 \ A_j^{10/3}} \frac{\partial P_j}{\partial y_j}$$

$$\frac{\partial S_{f_j}}{\partial Q_j} = \frac{n^2 Q_j P_j^{4/3}}{1.1041 \ A_j^{10/3}}$$

$$\frac{\partial \bar{y}A|_{j+1}}{\partial y_{j+1}} = A_{j+1}$$

$$\frac{\partial S_{f_{j+1}}}{\partial y_{j+1}} = - \frac{n^2 Q_{j+1} |Q_{j+1}| P_{j+1}^{4/3}}{0.66246 \ A_{j+1}^{13/3}} \frac{\partial A_{j+1}}{\partial y_{j+1}} + \frac{n^2 Q_{i,j+1} |Q_{j+1}| P_{j+1}^{1/3}}{1.65615 \ A_{j+1}^{10/3}} \frac{\partial P_{j+1}}{\partial y_{j+1}}$$

$$\frac{\partial S_{f_{j+1}}}{\partial Q_{j+1}} = \frac{n^2 \, Q_{j+1} \, P_{j+1}^{4/3}}{1.1041 \, A_{j+1}^{10/3}} \qquad [26]$$

The solution of equation [26] yields M values of dy and M values of dQ which are the changes to be applied to the approximate values of y and Q, respectively, in order to approach the solution. The improved values of y and Q are then substituted back into equation [24] and the cycle repeated as many times as required to approach the actual solution within the tolerance specified. In this manner, the solution progresses from one grid column to the next in the negative x direction until the solution is determined for the entire grid of points. This completes the computational procedure, and the upstream hydrograph can be produced by plotting the Q values along column i=1 as a function of time.

V. DISCUSSION OF THE RESULTS

The writer initially developed simplified computer models, utilizizing the implicit method, in an attempt to determine the full extent of the computational procedures available for use in forward and reverse flood routing. Using a simple sine wave input hydrograph and a rectangular channel, the writer attempted to compute a solution in each of the four possible directions in the x-t plane. In forward flood routing it is possible to use either the column procedure by computing the solution in the positive x direction or the row procedure by computing in the direction of positive time. In reverse flood routing the opposite is true. In this case the column procedure can be used by computing in the negative x direction or the row procedure by computing in the direction of negative time.

The writer was not successful in obtaining a stable solution using the column procedure in conjunction with forward flood routing. Solution instabilities were also encountered when reverse flood routing was attempted using the row procedure. The successful computer models, forward flood routing by the row procedure and reverse flood by the column procedure, were fully developed by the writer for application to natural channels. These two computer programs are included in Appendix A and Appendix B.

The fully developed computer models were tested using the 37 mile long reach from Reusen's Dam to Bent Creek on the James River. The flow period chosen was from March 30 to April 5, 1969, which is identical to that used in reference (4) for the same river reach. The lateral inflow per foot length of reach, QINFLO, was also identical to that used in reference (4); 0.00535 cfs/ft. An initial value of 0.035 was chosen for the Manning roughness number, n.

The cross section geometry and bed elevation was available for twelve stations along the reach, dividing the reach into eleven distance intervals. Table 1 lists the river mile, elevation, and bed slope at each of the twelve stations. The river cross section at each station was approximated by a triangle and two trapezoids as shown in Figure 5.

The forward flood routing computer model was tested first, using the discharge hydrograph at the dam as input. The discharge hydrograph is presented in Figure 6. The hydrograph consists of 165 hourly values of real data plus 35 hours of steady flow added to each end of the hydrograph.

The distance intervals and the time span chosen above resulted in a rectangular grid of points in the x-t plane of M = 200 rows and N = 12 columns. Boundary conditions were supplied by the dam hydrograph along column N = 1 and by a rating curve along column N = 12.

The dam hydrograph (Figure 6) was successfully routed through the reach resulting in the computed discharge hydrograph at Bent Creek as presented by Figure 7. Figure 7 compares the computed Bent Creek hydrograph with the actual Bent Creek hydrograph. It will be noted that the peaks of the hydrographs show a good agreement, while the valleys do not. This result is comparable to that attained in reference (4) for the same river reach and data input. Figure 8 presents

Table 1. River Station Features from Reusen's Dam to Bent Creek.

Station (N)	River Mile	Elevation ft. above MSL	Bed slope ft/ft
1	255.4	517.0	0.0015780
2	253.0	497.0	0.0011680
3	250.0	485.0	0.0005997
4	247.0	478.0	0.0006313
5	244.0	465.0	0.0008838
6	241.0	450.0	0.0008523
7	238.0	438.0	0.0007891
8	235.0	425.0	0.0007891
9	232.0	413.0	0.0007891
10	229.0	400.0	0.0008838
11	227.0	390.0	0.0008681
12	224.0	380.0	0.0007891

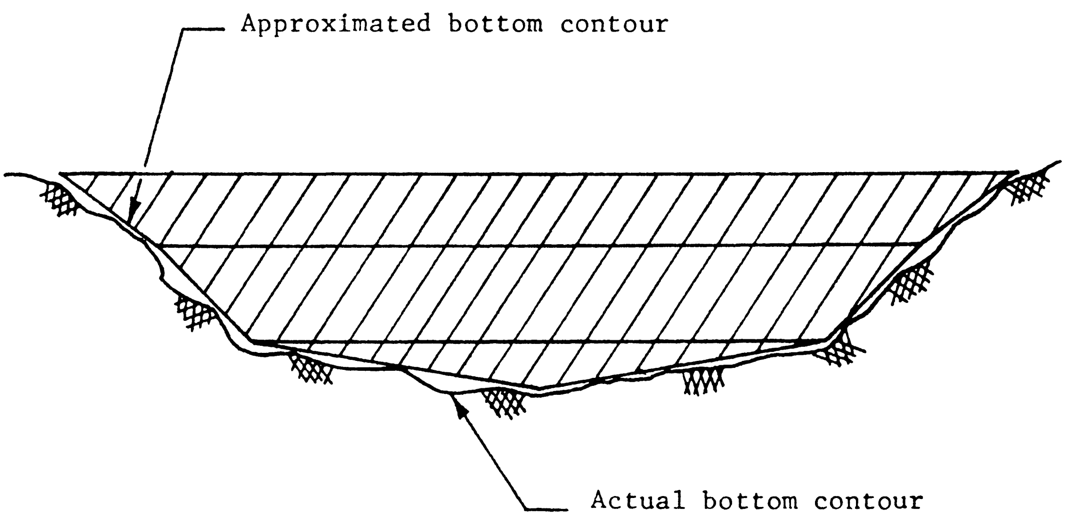

Figure 5. Method of approximation of station cross sections.

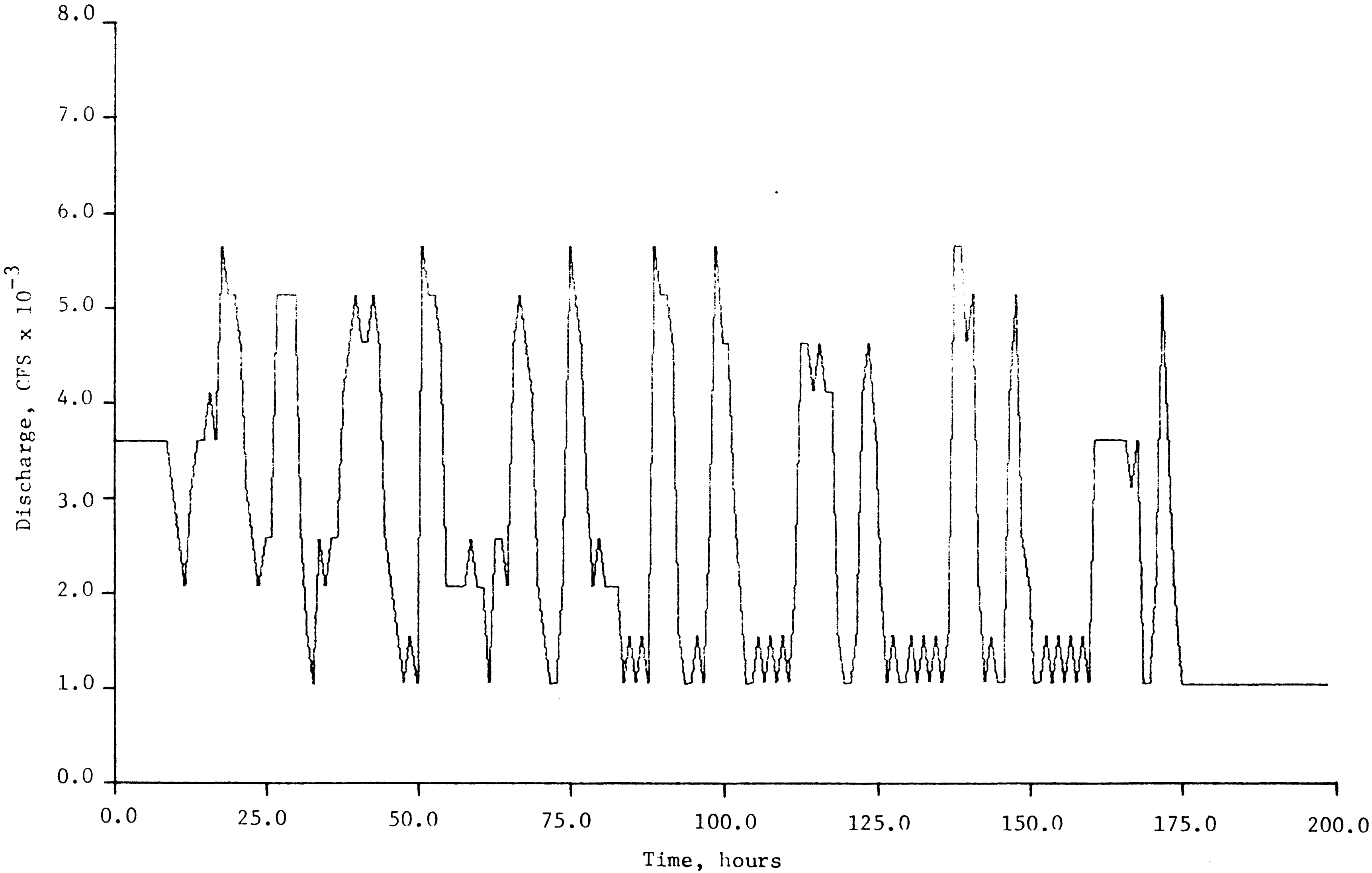

Figure 6. Actual discharge hydrograph at Reusen's Dam.

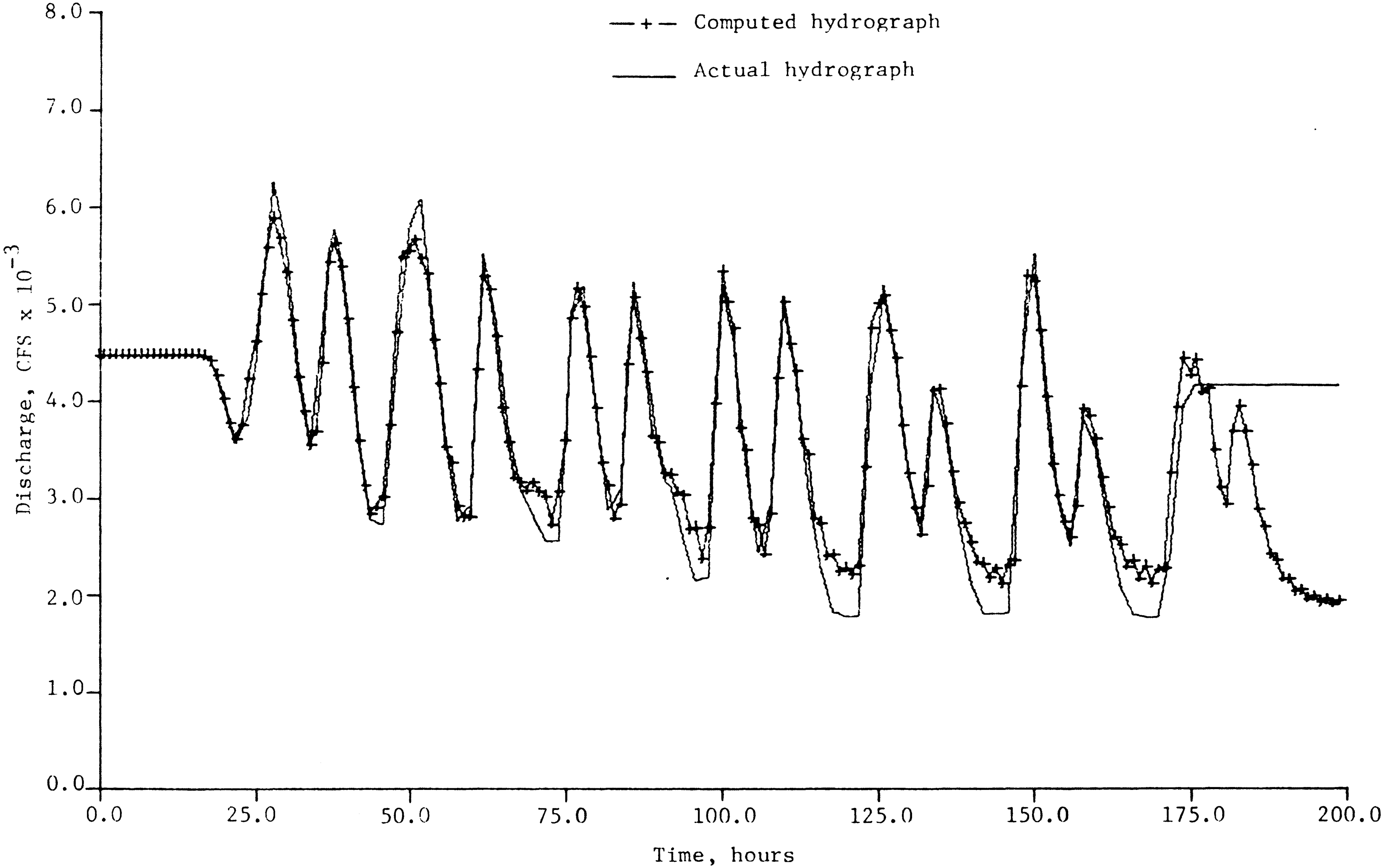

Figure 7. Computed discharge hydrograph and actual discharge hydrograph at Bent Creek.

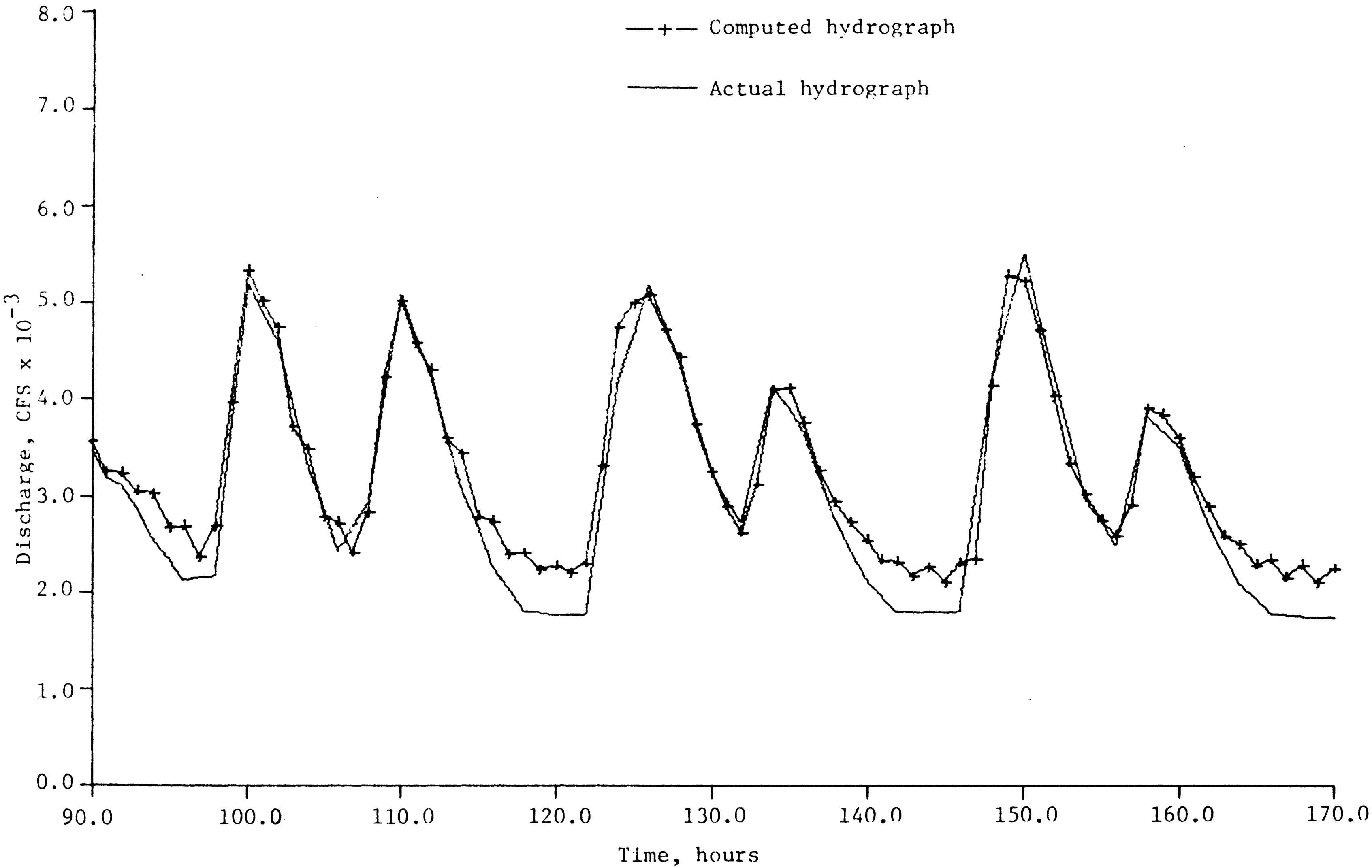

Figure 8. Replotted portion of Figure 7, on expanded time scale.

a portion of Figure 7 with the time scale expanded. The writer found it unnecessary to alter the Manning roughness number since the computed hydrograph was in good overall agreement with the actual hydrograph.

The reverse flood routing computer model was tested by using the computed Bent Creek hydrograph presented by Figure 7 as input. The N=12 column was initialized with the 200 discharge values corresponding to the computed Bent Creek discharge hydrograph. The corresponding depth values were determined from the Bent Creek rating curve. Boundary values of discharge were known along the M=1 and M=200 rows since these remained in steady flow regions of the grid. The various physical constants remained unchanged from that used in the forward flood routing computer model.

The reverse flood routing computer model produced the computed dam discharge hydrograph presented in Figure 9. In Figure 9 the computed hydrograph is compared to the actual hydrograph at the dam, where it will be noted that near perfect agreement was attained with the exception of a small deviation in the steady flow portion at the end of the hydrograph. In checking the individual values, the writer found that agreement was within 0.2%. Figure 10 presents a portion of the data from Figure 9 plotted on an expanded time scale. The time period selected was identical to that used in plotting Figure 8.

As an additional exercise for the reverse flood routing computer model, the actual Bent Creek hydrograph presented by Figure 11 was used as input. The iteration procedure converged to a solution for all columns except the last three near the dam. It was discovered that

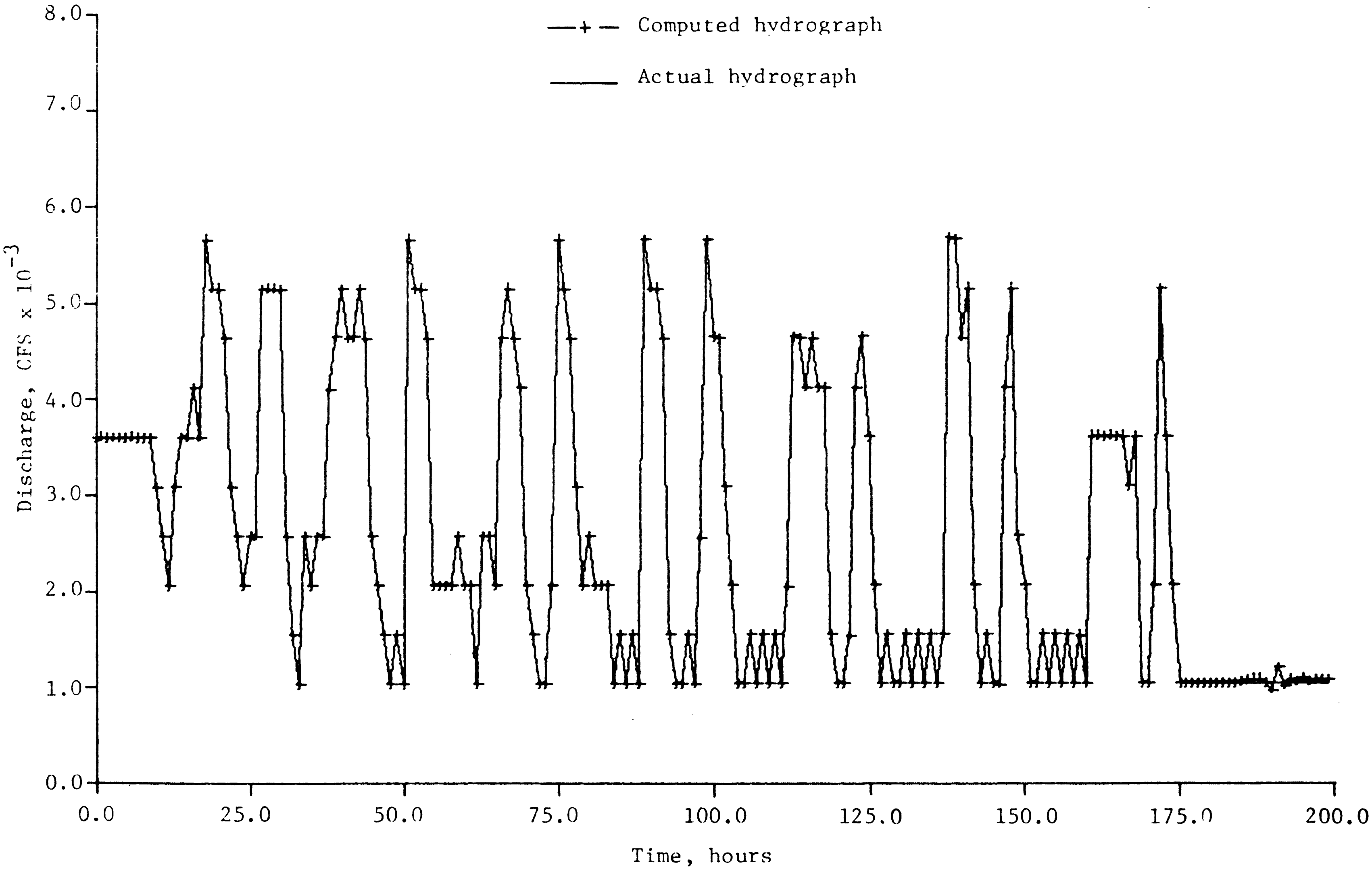

Figure 9. Computed discharge hydrograph and actual hydrograph at Reusen's Dam.

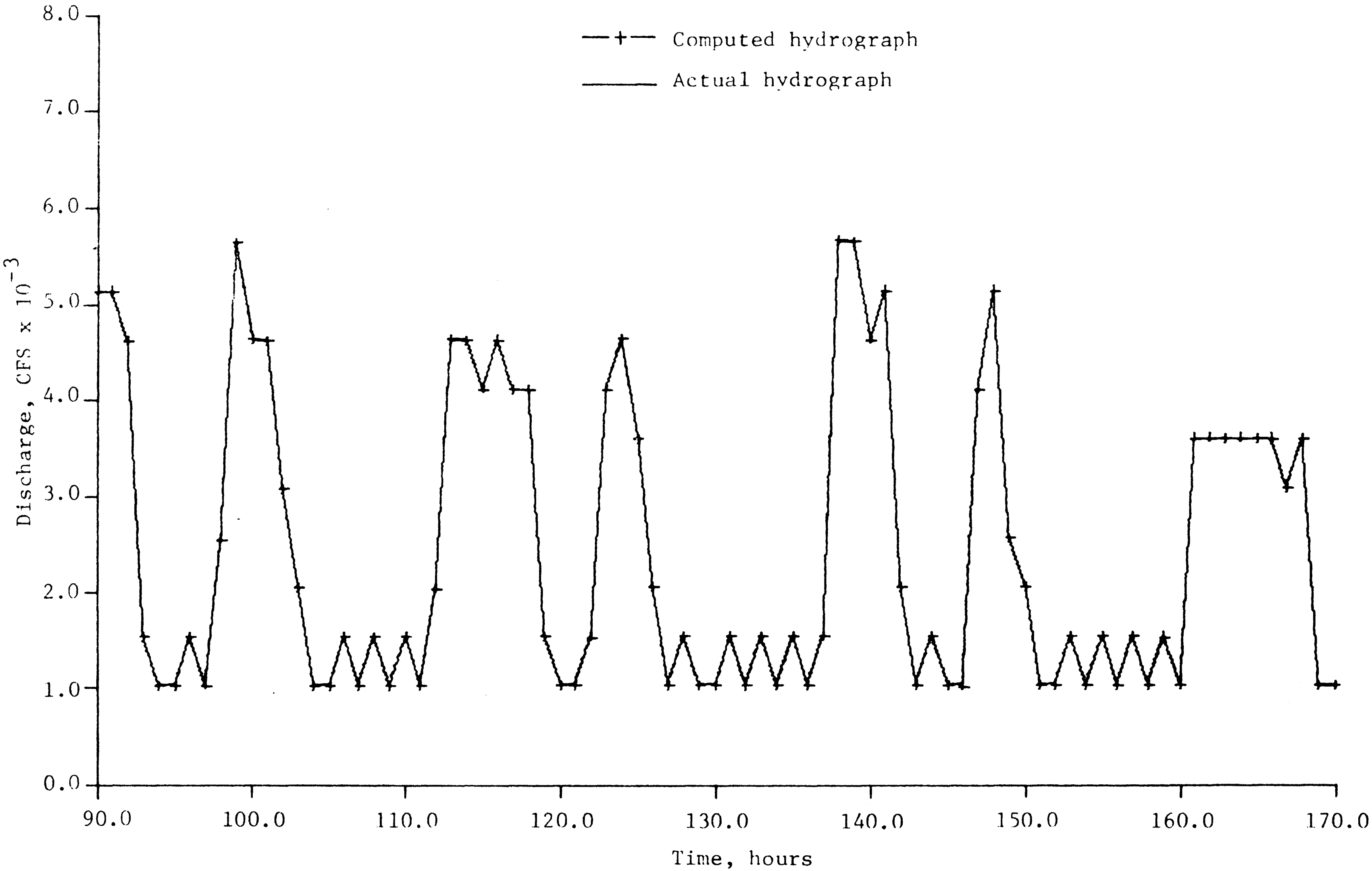

Figure 10. Replotted portion of Figure 9, on expanded time scale.

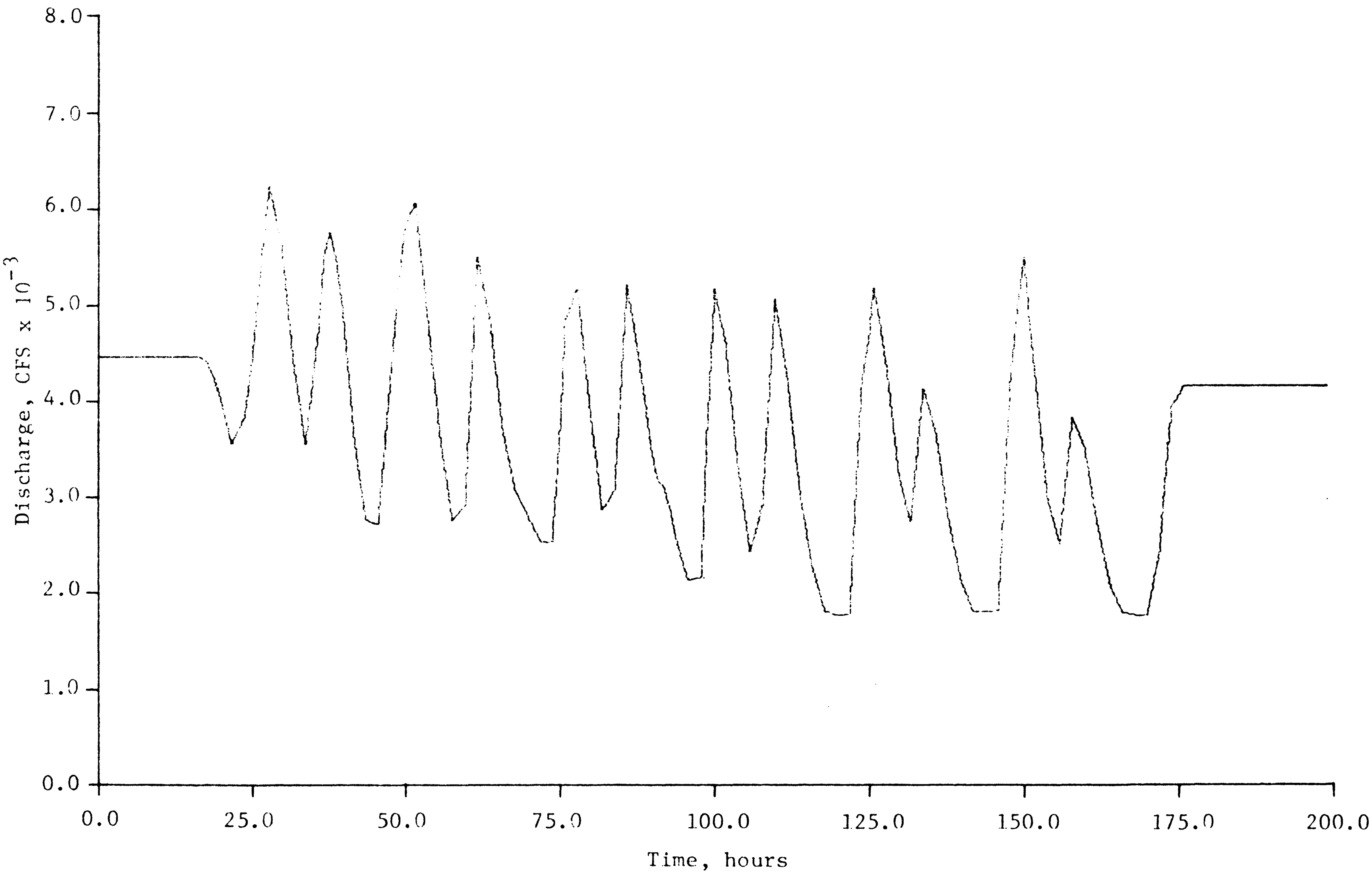

Figure 11. Actual discharge hydrograph at Bent Creek.

initial oscillations in the process of converging to a solution were resulting in negative discharge values for the lowest flow points in the valleys of the hydrograph. Since the computational scheme was not designed to handle negative discharges, divergence of the solution occurred. As will be noted by referring to Figure 7, the valleys of the actual Bent Creek hydrograph are somewhat lower than the computed version.

In order to provide additional assurance that the computer model would handle hydrographs at higher discharges, the writer initialized the model with the hypothetical discharge hydrograph shown in Figure 12. The resulting computed hydrograph at the dam is also presented by Figure 12.

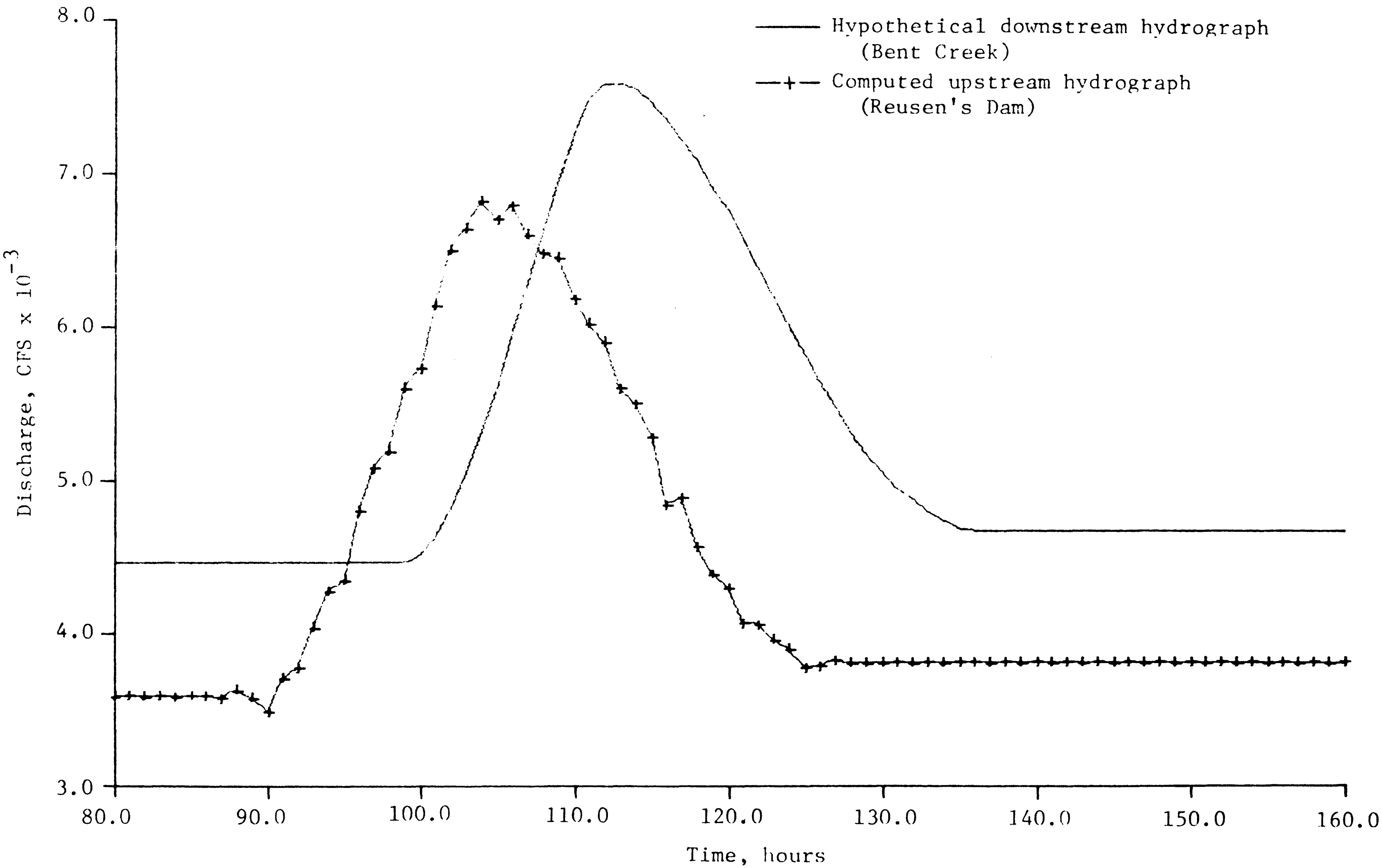

Figure 12. Hypothetical discharge hydrograph at Bent Creek and resulting computed discharge hydrograph at Reusen's Dam.

VI. CONCLUSIONS AND RECOMMENDATIONS

Strelkoff (9) presented an alternate form of the Saint-Venant equations in which discharge was the primary kinematic variable rather than velocity. The writer, while developing the continuity equation to an identical form to that presented by Strelkoff, preferred a different form of the momentum equation. The writer found, that by leaving the rate of change of momentum in total differential form, a very simple equation resulted (equation [11]). Unlike Strelkoff's version, it was possible to eliminate velocity from the momentum equation entirely. The final form of the governing equations utilized by the writer, equations [12] and [13], were found to be simple and concise, and could easily be expressed in finite difference form.

Computer models were developed for forward flood routing and reverse flood routing. The results indicated that the concept of reverse flood routing is feasible, with present evidence indicating that it can be accomplished just as easily as normal, forward flood routing. Although the writer was unable to develop computer models for forward flood routing using the column procedure and for reverse flood routing using the row procedure, no theoretical reasons are presented. It is recommended that additional efforts be made to obtain solutions using the above solution procedures, or that investigations be made to determine the reason convergence to a solution cannot be achieved. The ability to accomplish reverse flood routing using the row procedure would be of added benefit since the row procedure requires less computer storage and execution time than the column procedure.

The writer also recommends that a detailed mathematical study be made into the stability characteristics of the governing equations of continuity and momentum. Until the stability characteristics of the governing equations are fully known for a given computational scheme, there will continue to be unexpected and unexplained situations where convergence to a solution cannot be achieved.

REFERENCES

1. Amein, Michael and Fang, Ching S., "Implicit Flood Routing in Natural Channels," Journal of the Hydraulics Division, ASCE, Vol. 96, No. HY12, Proc. Paper 7773, Dec. 1970, pp. 2481-2055.

2. Baltzer, R. A. and Lai, Chintu, "Computer Simulation of Unsteady Flows in Waterways," Journal of the Hydraulics Division, ASCE, Vol. 94, No. HY4, Proc. Paper 6048, July, 1968, pp. 1083-1117.

3. Chow, V. T., Open Channel Hydraulics, McGraw-Hill Civil Engineering Series, McGraw-Hill Book Company, Inc., New York, 1959.

4. Contractor, D. N. and Wiggert, J. M., "Numerical Studies of Unsteady Flow in the James River," Water Resources Research Center, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, VPI-WRRC-Bull. 51, May 1972.

5. Garrison, J. M., Granju, Jean-Pierre P., and Price, J. T., "Unsteady Flow Simulation in Rivers and Reservoirs," Journal of the Hydraulics Division, ASCE, Vol. 95, No. HY5, Proc. Paper 6771, Sept. 1969, 1559-1576.

6. Gerald, C. F., Applied Numerical Analysis, Addison Wesley Publishing Company, Reading, Massachusetts, 1970.

7. Henderson, F. M., Open Channel Flow, Macmillan Series in Civil Engineering, The Macmillan Company, New York, 1971.

8. Stoker, S. S. Water Waves, Pure and Applied Mathematics Volume IV, Interscience Publishers, Inc., New York, 1957.

9. Strelkoff, Theodor, "Numerical Solution of Saint-Venant Equations," Journal of the Hydraulics Division, ASCE, Vol. 96, No. HY1, Proc. Paper. 7043, Jan., 1970, pp. 223-252.

APPENDIX A

```
C   FORWARD FLOOD ROUTING COMPUTER MODEL
C
C   WHERE:
C   Q = DISCHARGE
C   Y = DEPTH
C   A = CROSS SECTION AREA
C   YB = DISTANCE FROM WATER SURFACE TO CENTROID OF AREA, A
C   SF = FRICTION SLOPE
C   R = RESIDUES
C   CO = COEFFICIENTS (FORMS BANDED MATRIX)
C   DX = DISTANCE INCREMENT
C   SO = BED SLOPE
C   B = TOP WIDTH
C   WP = WETTED PERIMETER
C   WPY = RATE OF CHANGE OF WETTED PERIMETER WITH RESPECT TO DEPTH, Y
C   QU = UPSTREAM DISCHARGE HYDROGRAPH USED AS INPUT - 200 VALUES
C   YSEC & TOPW = STATION CROSS SECTION DATA, DESCRIBES CROSS SECTION
C   QDR & YDR = DOWNSTREAM RATING CURVE, DISCHARGE VS DEPTH
C   DT = TIME INTERVAL
C   RN = MANNING ROUGHNESS NUMBER
C   TOL = TOLERANCE VALUE IN PERCENT, USED TO TEST VALUES OF DQ AND DY
C   QINFLO = LATERAL INFLOW PER FOOT OF REACH
       COMMON Q(12,200),Y(12,200),A(12,2),YB(12,2),SF(12,2),R(24),CO(117)
      1,DX(11),SO(12),B(12),WP(12),WPY(12),QU(200),YSEC(12,4),TOPW(12,4),
      2QDR(12),YDR(12)
C   READ IN DOWNSTREAM RATING CURVE, DISCHARGE VS DEPTH
       DO 23 L=1,12
    24 FORMAT(F8.2)
       READ(5,24)YDR(L)
    23 READ(5,24)QDR(L)
C   READ IN CROSS SECTION DATA
       DO 52 I=1,12
```

```
      DO 53 M=1,4
   54 FORMAT(F7.2)
      READ(5,54)YSEC(I,M)
   55 FORMAT(F7.2)
   53 READ(5,55)TOPW(I,M)
   52 CONTINUE
      DT=3600.
      RN=.035
      TOL=.5
      QINFLO=.00535
      NC=1
      J=0
      CALL BEDDX
      CALL UPSTM(QINFLO)
      CALL AREA(J,RN,NC)
      NC=2
      DO 2 J=1,199
      K=0
C  INITIALIZE ROW J+1 WITH J VALUES, WITH EXCEPTION OF UPSTREAM BOUNDARY
C  VALUE, QU
      DO 3 I=1,12
      IF(I-1)4,5,4
    5 Q(1,J+1)=QU(J+1)
      GO TO 3
    4 Q(I,J+1)= Q(I,J)
    3 Y(I,J+1)=Y(I,J)
   17 K=K+1
C  CALCULATE ROW J+1 VALUES, A, B, YB, WP, WPY,  & SF
      CALL AREA(J,RN,NC)
      CALL RATE(YD,DIFFY,DIFFQ,J)
C  CALCULATE RESIDUES
      R(1)=Q(1,J+1)-QU(J+1)
```

```
      R(24)=Y(12,J+1)-YD
      DO 10 I=1,11
      M=2*I
      RC1=(Q(I+1,J)+Q(I+1,J+1)-Q(I,J)-Q(I,J+1))/(2.*DX(I))
      RC2=(A(I,2)+A(I+1,2)-A(I,1)-A(I+1,1))/(2.*DT)
      RC3=-QINFLO
      R(M)=RC1+RC2+RC3
      RM1=(YB(I,1)*A(I,1)+YB(I,2)*A(I,2)-YB(I+1,1)*A(I+1,1)-YB(I+1,2)*A(
     1I+1,2))/2.
      RM2=(SO(I)*A(I,1)+SO(I+1)*A(I+1,1)+SO(I)*A(I,2)+SO(I+1)*A(I+1,2))*
     1DX(I)/4.
      RM3=-DX(I)*(SF(I,1)*A(I,1)+SF(I+1,1)*A(I+1,1)+SF(I,2)*A(I,2)+SF(I+
     11,2)*A(I+1,2))/4.
      RM4=-(Q(I+1,J+1)*ABS(Q(I+1,J+1))/A(I+1,2)-Q(I,J)*ABS(Q(I,J))/A(I,1
     1))/32.17
      R(M+1)=RM1+RM2+RM3+RM4
   10 CONTINUE
C  CHANGE RESIDUE SIGN
      DO 11 M=1,24
   11 R(M)=-R(M)
C  CALCULATION OF COEFFICIENTS, CO, FOR BANDED MATRIX
      CO(1)=1.
      CO(2)=0.0
      CO(113)=-DIFFY/DIFFQ
      CO(114)=1.0
      DO 12 I=1,11
      CO(I*10-7)=0.0
      CO(I*10-6)=-1./(2.*DX(I))
      CO(I*10-5)=B(I)/(2.*DT)
      CO(I*10-4)=1./(2.*DX(I))
      CO(I*10-3)=B(I+1)/(2.*DT)
      CO(I*10-2)=-DX(I)*RN*RN*Q(I,J+1)*WP(I)**(4./3.)/(4.4164*A(I,2)**(7
```

```
     1./3.))
      COJ1=A(I,2)/2.+DX(I)*B(I)*(SO(I)-SF(I,2))/4.
      COJ2=DX(I)*RN*RN*Q(I,J+1)*ABS(Q(I,J+1))
      COJ3=WP(I)**(4./3.)*B(I)/(2.64984*A(I,2)**(10./3.))
      COJ4=-WP(I)**(1./3.)*WPY(I)/(6.62460*A(I,2)**(7./3.))
      CO(I*10-1)=COJ1+COJ2*(COJ3+COJ4)
      COJ5=-DX(I)*RN*RN*Q(I+1,J+1)*WP(I+1)**(4./3.)/(4.4164*A(I+1,2)**(7
     1./3.))
      COJ6=-2.*Q(I+1,J+1)/(32.17*A(I+1,2))
      CO(I*10)=COJ5+COJ6
      COJ7=-A(I+1,2)/2.+DX(I)*B(I+1)*(SO(I+1)-SF(I+1,2))/4.
      COJ8=DX(I)*RN*RN*Q(I+1,J+1)*ABS(Q(I+1,J+1))
      COJ9=WP(I+1)**(4./3.)*B(I+1)/(2.64984*A(I+1,2)**(10./3.))
      COJ10=-WP(I+1)**(1./3.)*WPY(I+1)/(6.62460*A(I+1,2)**(7./3.))
      COJ11=Q(I+1,J+1)*ABS(Q(I+1,J+1))*B(I+1)/(32.17*A(I+1,2)*A(I+1,2))
      CO(I*10+1)=COJ7+COJ8*(COJ9+COJ10)+COJ11
   12 CO(I*10+2)=0.0
C  SOLVE MATRIX
      CALL GELB(R,CO,24,1,2,2,5.0E-07,IER)
   40 FORMAT(5X,3HK= ,I4,5X,5HIER= ,I4,5X,3HJ= ,I4)
      WRITE(6,40)K,IER,J
      IF(K-10)51,70,70
C  SEE IF ERROR VALUES, DQ & DY, ARE LESS THAT SPECIFIED TOLERANCE
   51 DO 14 I=1,12
      ERRORQ=R(2*I-1)/Q(I,J+1)
      IF(ABS(ERRORQ)-TOL/100.)15,15,16
   15 ERRORY=R(2*I)/Y(I,J+1)
      IF(ABS(ERRORY)-TOL/100.)14,14,16
   14 CONTINUE
      GO TO 21
C  COMPUTE K+1 VALUES OF DISCHARGE AND DEPTH
   16 DO 20 I=1,12
```

```
      Q(I,J+1)=Q(I,J+1)+R(2*I-1)
   20 Y(I,J+1)=Y(I,J+1)+R(2*I)
      GO TO 17
C  COMPUTE FINAL VALUES OF DISCHARGE AND DEPTH
   21 DO 22 I=1,12
      Q(I,J+1)=Q(I,J+1)+R(2*I-1)
   22 Y(I,J+1)=Y(I,J+1)+R(2*I)
C  CALCULATE FINAL VALUES OF A, B, YB, WP, WPY, & SF FOR ROW J+1
      CALL AREA(J,RN,NC)
C  MOVE FORWARD VALUES OF A, YB & SF
      DO 18 I=1,12
      A(I,1)=A(I,2)
      YB(I,1)=YB(I,2)
   18 SF(I,1)=SF(I,2)
    2 CONTINUE
C  PRINT FINAL VALUES OF Q AND Y FOR GRID POINTS
      DO 60 J=1,200
   61 FORMAT(2X,12F10.2)
      WRITE(6,61)Q(1,J),Q(2,J),Q(3,J),Q(4,J),Q(5,J),Q(6,J),Q(7,J),Q(8,J)
     1,Q(9,J),Q(10,J),Q(11,J),Q(12,J)
   62 FORMAT(2X,12F10.4)
      WRITE(6,62)Y(1,J),Y(2,J),Y(3,J),Y(4,J),Y(5,J),Y(6,J),Y(7,J),Y(8,J)
     1,Y(9,J),Y(10,J),Y(11,J),Y(12,J)
   60 CONTINUE
   70 CONTINUE
      STOP
      END
```

```
      SUBROUTINE BEDDX
C   SUBROUTINE "BEDDX" READS IN BED SLOPE VALUES, SO, FOR EACH STATION,
C   AND, READS IN THE DISTANCE INCREMENTS, DX
      COMMON Q(12,200),Y(12,200),A(12,2),YB(12,2),SF(12,2),R(24),CO(117)
     1,DX(11),SO(12),B(12),WP(12),WPY(12),QU(200),YSEC(12,4),TOPW(12,4),
     2QDR(12),YDR(12)
      DO 101 L=1,12
  100 FORMAT(F10.8)
  101 READ(5,100)SO(L)
      DO 102 L=1,11
  103 FORMAT(F7.1)
  102 READ(5,103)DX(L)
  902 FORMAT(2X,'BED SLOPE (1-12) ',12F9.7)
      WRITE(6,902)SO(1),SO(2),SO(3),SO(4),SO(5),SO(6),SO(7),SO(8),SO(9),
     1SO(10),SO(11),SO(12)
  903 FORMAT(2X,'DX (1-11) ',11F9.1)
      WRITE(6,903)DX(1),DX(2),DX(3),DX(4),DX(5),DX(6),DX(7),DX(8),DX(9),
     1DX(10),DX(11)
      RETURN
      END
```

```
      SUBROUTINE UPSTM(QINFLO)
C   SUBROUTINE "UPSTM" READS IN UPSTREAM DISCHARGE HYDROGRAPH, QU VS TIME
C   AND, INITIALIZES THE M = 1 ROW WITH VALUES OF Q AND Y
      COMMON Q(12,200),Y(12,200),A(12,2),YB(12,2),SF(12,2),R(24),CO(117)
     1,DX(11),SO(12),B(12),WP(12),WPY(12),QU(200),YSEC(12,4),TOPW(12,4),
     2QDR(12),YDR(12)
      DO 200 M=1,200
  204 FORMAT(F8.2)
  200 READ(5,204)QU(M)
      Q(1,1)=3591.
      DO 201 I=1,11
  201 Q(I+1,1)=Q(I,1)+DX(I)*QINFLO
      DO 202 I=1,12
  203 FORMAT(F7.4)
  202 READ(5,203)Y(I,1)
      RETURN
      END
```

```
      SUBROUTINE RATE(YD,DIFFY,DIFFQ,J)
C   SUBROUTINE "RATE" USES CURRENT VALUE OF Q(12,J+1), DETERMINES
C   CORRESPONDING DEPTH, Y(12,J+1) FROM DOWNSTREAM RATING CURVE
      COMMON Q(12,200),Y(12,200),A(12,2),YB(12,2),SF(12,2),R(24),CO(117)
     1,DX(11),SO(12),B(12),WP(12),WPY(12),QU(200),YSEC(12,4),TOPW(12,4),
     2QDR(12),YDR(12)
      DO 400 L=1,12
      IF(QDR(L)-Q(12,J+1))400,400,402
  400 CONTINUE
  402 DIFFQ=QDR(L)-QDR(L-1)
      DIFFY=YDR(L)-YDR(L-1)
      DFQ=QDR(L)-Q(12,J+1)
      DFY=DFQ*DIFFY/DIFFQ
      YD=YDR(L)-DFY
      RETURN
      END
```

```
      SUBROUTINE AREA(J,RN,NC)
C   SUBROUTINE "AREA" COMPUTES CURRENT VALUES OF A, B, YB, WP, WPY, & SF
      COMMON Q(12,200),Y(12,200),A(12,2),YB(12,2),SF(12,2),R(24),CO(117)
     1,DX(11),SO(12),B(12),WP(12),WPY(12),QU(200),YSEC(12,4),TOPW(12,4),
     2QDR(12),YDR(12)
      DO 302 I=1,12
      HTOPW2=TOPW(I,2)/2.
      HTOPW3=TOPW(I,3)/2.
      HTOPW4=TOPW(I,4)/2.
      YSEC32=YSEC(I,3)-YSEC(I,2)
      YSEC43=YSEC(I,4)-YSEC(I,3)
      WP1=2.*SQRT(YSEC(I,2)**2.+HTOPW2**2.)
      WP2=2.*SQRT(YSEC32**2.+(HTOPW3-HTOPW2)**2.)
      A1=HTOPW2*YSEC(I,2)
      A2=HTOPW2*YSEC32
      A3=HTOPW3*YSEC32
      SLOPE1=YSEC(I,2)/HTOPW2
      SLOPE2=YSEC32/(HTOPW3-HTOPW2)
      SLOPE3=YSEC43/(HTOPW4-HTOPW3)
      VY=Y(I,J+1)
      IF(YSEC(I,2)-VY)305,304,304
  304 HX=VY/SLOPE1
      A(I,NC)=HX*VY
      YB(I,NC)=VY/3.
      WP(I)=2.*SQRT(VY**2.+HX**2.)
      WPY(I)=2.*SQRT(SLOPE1**(-2.)+1.)
      GO TO 315
  305 IF(YSEC(I,3)-VY)307,306,306
  306 VYSEC2=VY-YSEC(I,2)
      HXT1=VYSEC2/SLOPE2
      HX=HTOPW2+HXT1
      AT2=HTOPW2*VYSEC2
```

```
      AT3=HX*VYSEC2
      A(I,NC)=A1+AT2+AT3
      WPT1=2.*SQRT(VYSEC2**2.+HXT1**2.)
      WP(I)=WP1+WPT1
      MOM1=A1*(VY-YSEC(I,2)*2./3.)
      MOMT2=AT2*(VYSEC2*2./3.)
      MOMT3=AT3*(VYSEC2/3.)
      TMOM=MOM1+MOMT2+MOMT3
      YB(I,NC)=TMOM/A(I,NC)
      WPY(I)=2.*SQRT(SLOPE2**(-2.)+1.)
      GO TO 315
  307 VYSEC3=VY-YSEC(I,3)
      HXT2=VYSEC3/SLOPE3
      HX=HTOPW3+HXT2
      AT4=HTOPW3*VYSEC3
      AT5=HX*VYSEC3
      A(I,NC)=A1+A2+A3+AT4+AT5
      WPT2=2.*SQRT(VYSEC3**2.+HXT2**2.)
      WP(I)=WP1+WP2+WPT2
      MOM1=A1*(VY-YSEC(I,2)*2./3.)
      MOM2=A2*(YSEC32*2./3.+VYSEC3)
      MOM3=A3*(YSEC32/3.+VYSEC3)
      MOMT4=AT4*(VYSEC3*2./3.)
      MOMT5=AT5*(VYSEC3/3.)
      TMOM=MOM1+MOM2+MOM3+MOMT4+MOMT5
      YB(I,NC)=TMOM/A(I,NC)
      WPY(I)=2.*SQRT(SLOPE3**(-2.)+1.)
  315 B(I)=2.*HX
  302 SF(I,NC)=RN*RN*Q(I,J+1)*ABS(Q(I,J+1))*WP(I)**(4./3.)/(2.2082*A(I,N
     1C)**(10./3.))
      RETURN
      END
```

```
C
C     ..................................................................
C
C        SUBROUTINE GELB
C
C        PURPOSE
C           TO SOLVE A SYSTEM OF SIMULTANEOUS LINEAR EQUATIONS WITH A
C           COEFFICIENT MATRIX OF BAND STRUCTURE.
C
C        USAGE
C           CALL GELB(R,A,M,N,MUD,MLD,EPS,IER)
C
C        DESCRIPTION OF PARAMETERS
C           R      - M BY N RIGHT HAND SIDE MATRIX (DESTROYED).
C                    ON RETURN R CONTAINS THE SOLUTION OF THE EQUATIONS.
C           A      - M BY M COEFFICIENT MATRIX WITH BAND STRUCTURE
C                    (DESTROYED).
C           M      - THE NUMBER OF EQUATIONS IN THE SYSTEM.
C           N      - THE NUMBER OF RIGHT HAND SIDE VECTORS.
C           MUD    - THE NUMBER OF UPPER CODIAGONALS (THAT MEANS
C                    CODIAGONALS ABOVE MAIN DIAGONAL).
C           MLD    - THE NUMBER OF LOWER CODIAGONALS (THAT MEANS
C                    CODIAGONALS BELOW MAIN DIAGONAL).
C           EPS    - AN INPUT CONSTANT WHICH IS USED AS RELATIVE
C                    TOLERANCE FOR TEST ON LOSS OF SIGNIFICANCE.
C           IER    - RESULTING ERROR PARAMETER CODED AS FOLLOWS
C                    IER=0  - NO ERROR,
C                    IER=-1 - NO RESULT BECAUSE OF WRONG INPUT PARAME-
C                             TERS M,MUD,MLD OR BECAUSE OF PIVOT ELEMENT
C                             AT ANY ELIMINATION STEP EQUAL TO 0,
C                    IER=K  - WARNING DUE TO POSSIBLE LOSS OF SIGNIFI-
C                             CANCE INDICATED AT ELIMINATION STEP K+1,
```

```
C                              WHERE PIVOT ELEMENT WAS LESS THAN OR
C                              EQUAL TO THE INTERNAL TOLERANCE EPS TIMES
C                              ABSOLUTELY GREATEST ELEMENT OF MATRIX A.
C
C        REMARKS
C           BAND MATRIX A IS ASSUMED TO BE STORED ROWWISE IN THE FIRST
C           ME SUCCESSIVE STORAGE LOCATIONS OF TOTALLY NEEDED MA
C           STORAGE LOCATIONS, WHERE
C             MA=M*MC-ML*(ML+1)/2    AND    ME=MA-MU*(MU+1)/2    WITH
C             MC=MIN(M,1+MUD+MLD),  ML=MC-1-MLD,  MU=MC-1-MUD.
C           RIGHT HAND SIDE MATRIX R IS ASSUMED TO BE STORED COLUMNWISE
C           IN N*M SUCCESSIVE STORAGE LOCATIONS. ON RETURN SOLUTION
C           MATRIX R IS STORED COLUMNWISE TOO.
C           INPUT PARAMETERS M, MUD, MLD SHOULD SATISFY THE FOLLOWING
C           RESTRICTIONS     MUD NOT LESS THAN ZERO
C                            MLD NOT LESS THAN ZERO
C                            MUD+MLD NOT GREATER THAN 2*M-2.
C           NO ACTION BESIDES ERROR MESSAGE IER=-1 TAKES PLACE IF THESE
C           RESTRICTIONS ARE NOT SATISFIED.
C           THE PROCEDURE GIVES RESULTS IF THE RESTRICTIONS ON INPUT
C           PARAMETERS ARE SATISFIED AND IF PIVOT ELEMENTS AT ALL
C           ELIMINATION STEPS ARE DIFFERENT FROM 0. HOWEVER WARNING
C           IER=K - IF GIVEN - INDICATES POSSIBLE LOSS OF SIGNIFICANCE.
C           IN CASE OF A WELL SCALED MATRIX A AND APPROPRIATE TOLERANCE
C           EPS, IER=K MAY BE INTERPRETED THAT MATRIX A HAS THE RANK K.
C           NO WARNING IS GIVEN IF MATRIX A HAS NO LOWER CODIAGONAL.
C
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C           NONE
C
C        METHOD
C           SOLUTION IS DONE BY MEANS OF GAUSS ELIMINATION WITH
```

```
C            COLUMN PIVOTING ONLY, IN ORDER TO PRESERVE BAND STRUCTURE
C            IN REMAINING COEFFICIENT MATRICES.
C
C     ..................................................................
C
      SUBROUTINE GELB(R,A,M,N,MUD,MLD,EPS,IER)
C
C
      DIMENSION R(24),A(117)
C
C     TEST ON WRONG INPUT PARAMETERS
      IF(MLD)47,1,1
    1 IF(MUD)47,2,2
    2 MC=1+MLD+MUD
      IF(MC+1-M-M)3,3,47
C
C     PREPARE INTEGER PARAMETERS
C        MC=NUMBER OF COLUMNS IN MATRIX A
C        MU=NUMBER OF ZEROS TO BE INSERTED IN FIRST ROW OF MATRIX A
C        ML=NUMBER OF MISSING ELEMENTS IN LAST ROW OF MATRIX A
C        MR=INDEX OF LAST ROW IN MATRIX A WITH MC ELEMENTS
C        MZ=TOTAL NUMBER OF ZEROS TO BE INSERTED IN MATRIX A
C        MA=TOTAL NUMBER OF STORAGE LOCATIONS NECESSARY FOR MATRIX A
C        NM=NUMBER OF ELEMENTS IN MATRIX R
    3 IF(MC-M)5,5,4
    4 MC=M
    5 MU=MC-MUD-1
      ML=MC-MLD-1
      MR=M-ML
      MZ=(MU*(MU+1))/2
      MA=M*MC-(ML*(ML+1))/2
      NM=N*M
```

```
C
C     MOVE ELEMENTS BACKWARD AND SEARCH FOR ABSOLUTELY GREATEST ELEMENT
C     (NOT NECESSARY IN CASE OF A MATRIX WITHOUT LOWER CODIAGONALS)
      IER=0
      PIV=0.
      IF(MLD)14,14,6
    6 JJ=MA
      J=MA-MZ
      KST=J
      DO 9 K=1,KST
      TB=A(J)
      A(JJ)=TB
      TB=ABS(TB)
      IF(TB-PIV)8,8,7
    7 PIV=TB
    8 J=J-1
    9 JJ=JJ-1
C
C     INSERT ZEROS IN FIRST MU ROWS (NOT NECESSARY IN CASE MZ=0)
      IF(MZ)14,14,10
   10 JJ=1
      J=1+MZ
      IC=1+MUD
      DO 13 I=1,MU
      DO 12 K=1,MC
      A(JJ)=0.
      IF(K-IC)11,11,12
   11 A(JJ)=A(J)
      J=J+1
   12 JJ=JJ+1
   13 IC=IC+1
C
```

```
C      GENERATE TEST VALUE FOR SINGULARITY
   14 TOL=EPS*PIV
C
C
C      START DECOMPOSITION LOOP
      KST=1
      IDST=MC
      IC=MC-1
      DO 38 K=1,M
      IF(K-MR-1)16,16,15
   15 IDST=IDST-1
   16 ID=IDST
      ILR=K+MLD
      IF(ILR-M)18,18,17
   17 ILR=M
   18 II=KST
C
C      PIVOT SEARCH IN FIRST COLUMN (ROW INDEXES FROM I=K UP TO I=ILR)
      PIV=0.
      DO 22 I=K,ILR
      TB=ABS(A(II))
      IF(TB-PIV)20,20,19
   19 PIV=TB
      J=I
      JJ=II
   20 IF(I-MR)22,22,21
   21 ID=ID-1
   22 II=II+ID
C
C      TEST ON SINGULARITY
      IF(PIV)47,47,23
   23 IF(IER)26,24,26
```

```
   24 IF(PIV-TOL)25,25,26
   25 IER=K-1
   26 PIV=1./A(JJ)
C
C        PIVOT ROW REDUCTION AND ROW INTERCHANGE IN RIGHT HAND SIDE R
         ID=J-K
         DO 27 I=K,NM,M
         II=I+ID
         TB=PIV*R(II)
         R(II)=R(I)
   27 R(I)=TB
C
C        PIVOT ROW REDUCTION AND ROW INTERCHANGE IN COEFFICIENT MATRIX A
         II=KST
         J=JJ+IC
         DO 28 I=JJ,J
         TB=PIV*A(I)
         A(I)=A(II)
         A(II)=TB
   28 II=II+1
C
C        ELEMENT REDUCTION
         IF(K-ILR)29,34,34
   29 ID=KST
         II=K+1
         MU=KST+1
         MZ=KST+IC
         DO 33 I=II,ILR
C
C        IN MATRIX A
         ID=ID+MC
         JJ=I-MR-1
```

```
      IF(JJ)31,31,30
   30 ID=ID-JJ
   31 PIV=-A(ID)
      J=ID+1
      DO 32 JJ=MU,MZ
      A(J-1)=A(J)+PIV*A(JJ)
   32 J=J+1
      A(J-1)=0.
C
C     IN MATRIX R
      J=K
      DO 33 JJ=I,NM,M
      R(JJ)=R(JJ)+PIV*R(J)
   33 J=J+M
   34 KST=KST+MC
      IF(ILR-MR)36,35,35
   35 IC=IC-1
   36 ID=K-MR
      IF(ID)38,38,37
   37 KST=KST-ID
   38 CONTINUE
C     END OF DECOMPOSITION LOOP
C
C
C     BACK SUBSTITUTION
      IF(MC-1)46,46,39
   39 IC=2
      KST=MA+ML-MC+2
      II=M
      DO 45 I=2,M
      KST=KST-MC
      II=II-1
```

```
      J=II-MR
      IF(J)41,41,40
   40 KST=KST+J
   41 DO 43 J=II,NM,M
      TB=R(J)
      MZ=KST+IC-2
      ID=J
      DO 42 JJ=KST,MZ
      ID=ID+1
   42 TB=TB-A(JJ)*R(ID)
   43 R(J)=TB
      IF(IC-MC)44,45,45
   44 IC=IC+1
   45 CONTINUE
   46 RETURN
C
C
C     ERROR RETURN
   47 IER=-1
      RETURN
      END
```

APPENDIX B

```
C   REVERSE FLOOD ROUTING COMPUTER MODEL
C
C   WHERE:
C   Q = DISCHARGE
C   Y = DEPTH
C   A = CROSS SECTION AREA
C   YB = DISTANCE FROM WATER SURFACE TO CENTROID OF AREA, A
C   SF = FRICTION SLOPE
C   R = RESIDUES
C   CO = COEFFICIENTS (FORMS BANDED MATRIX)
C   DX = DISTANCE INCREMENT
C   SO = BED SLOPE
C   B = TOP WIDTH
C   WP = WETTED PERIMETER
C   WPY = RATE OF CHANGE OF WETTED PERIMETER WITH RESPECT TO DEPTH, Y
C   YSEC & TOPW = STATION CROSS SECTION DATA, DESCRIBES CROSS SECTION
C   QDR & YDR = DOWNSTREAM RATING CURVE, DISCHARGE VS DEPTH
C   DT = TIME INTERVAL
C   RN = MANNING ROUGHNESS NUMBER
C   TOL = TOLERANCE VALUE IN PERCENT, USED TO TEST VALUES OF DQ AND DY
C   QINFLO = LATERAL INFLOW PER FOOT OF REACH
        COMMON Q(12,200),Y(12,200),A(2,200),YB(2,200),SF(2,200),R(400),CO(
       11997),DX(11),SO(12),WP(200),B(200),WPY(200),YSEC(12,4),TOPW(12,4),
C   READ IN CROSS SECTION DATA
       2QDR(12),YDR(12)
        DO 52 I=1,12
        DO 53 M=1,4
    54  FORMAT(F7.2)
        READ(5,54)YSEC(I,M)
    55  FORMAT(F7.2)
    53  READ(5,55)TOPW(I,M)
    52  CONTINUE
```

```
      DT=3600.
      RN=.035
      TOL=.5
      QINFLO=.00535
      DQDX=0.0
      NC=2
      I=12
      CALL BEDDX
      CALL DWNSTM
      CALL AREA(I,RN,NC)
      NC=1
      DO 2 II=1,11
      I=12-II
      K=0
      DQDX=DX(I)*QINFLO
C  INITIALIZE COLUMN I WITH I+1 VALUES OF DISCHARGE AND DEPTH
      DO 3 J=1,200
      Q(I,J)=Q(I+1,J)
    3 Y(I,J)=Y(I+1,J)
   17 K=K+1
C  CALCULATE COLUMN I VALUES, A, B, YB, WP, WPY, & SF
      CALL AREA(I,RN,NC)
C  CALCULATE RESIDUES, R
      R(1)=Q(I,1)-(Q(I+1,1)-DQDX)
      R(400)=Q(I,200)-(Q(I+1,200)-DQDX)
      DO 10 J=1,199
      M=2*J
      RC1=(Q(I+1,J)+Q(I+1,J+1)-Q(I,J)-Q(I,J+1))/(2.*DX(I))
      RC2=(A(1,J+1)+A(2,J+1)-A(1,J)-A(2,J))/(2.*DT)
      RC3=-QINFLO
      R(M)=RC1+RC2+RC3
      RM1=(YB(1,J)*A(1,J)+YB(1,J+1)*A(1,J+1)-YB(2,J)*A(2,J)-YB(2,J+1)*A(
```

```
     12,J+1))/2.
      RM2=(SO(I)*A(1,J)+SO(I+1)*A(2,J)+SO(I)*A(1,J+1)+SO(I+1)*A(2,J+1))*
     1DX(I)/4.
      RM3=-DX(I)*(SF(1,J)*A(1,J)+SF(2,J)*A(2,J)+SF(1,J+1)*A(1,J+1)+SF(2,
     1J+1)*A(2,J+1))/4.
      RM4=-(Q(I+1,J+1)*ABS(Q(I+1,J+1))/A(2,J+1)-Q(I,J)*ABS(Q(I,J))/A(1,J
     1))/32.17
      R(M+1)=RM1+RM2+RM3+RM4
   10 CONTINUE
C  CHANGE RESIDUE SIGN
      DO 11 M=1,400
   11 R(M)=-R(M)
C  CALCULATION OF COEFFICIENTS, CO, FOR BANDED MATRIX
      CO(1)=0.0
      CO(2)=1.
      CO(1993)=0.0
      CO(1994)=1.
      DO 12 J=1,199
      CO(J*10-7)=0.0
      CO(J*10-6)=-B(J)/(2.*DT)
      CO(J*10-5)=-1./(2.*DX(I))
      CO(J*10-4)=B(J+1)/(2.*DT)
      CO(J*10-3)=-1./(2.*DX(I))
      COJ1=A(1,J)/2.+DX(I)*B(J)*(SO(I)-SF(1,J))/4.
      COJ2=DX(I)*RN*RN*Q(I,J)*ABS(Q(I,J))
      COJ3=WP(J)**(4./3.)*B(J)/(2.64984*A(1,J)**(10./3.))
      COJ4=-WP(J)**(1./3.)*WPY(J)/(6.62460*A(1,J)**(7./3.))
      COJ5=-Q(I,J)*ABS(Q(I,J))*B(J)/(32.17*A(1,J)*A(1,J))
      CO(J*10-2)=COJ1+COJ2*(COJ3+COJ4)+COJ5
      COJ6=-DX(I)*RN*RN*Q(I,J)*WP(J)**(4./3.)/(4.4164*A(1,J)**(7./3.))
      COJ7=2.*Q(I,J)/(32.17*A(1,J))
      CO(J*10-1)=COJ6+COJ7
```

```
      COJ8=A(1,J+1)/2.+DX(I)*B(J+1)*(SO(I)-SF(1,J+1))/4.
      COJ9=DX(I)*RN*RN*Q(I,J+1)*ABS(Q(I,J+1))
      COJ10=WP(J+1)**(4./3.)*B(J+1)/(2.64984*A(1,J+1)**(10./3.))
      COJ11=-WP(J+1)**(1./3.)*WPY(J+1)/(6.62460*A(1,J+1)**(7./3.))
      CO(J*10)=COJ8+COJ9*(COJ10+COJ11)
      CO(J*10+1)=-DX(I)*RN*RN*Q(I,J+1)*WP(J+1)**(4./3.)/(4.4164*A(1,J+1)
     1**(7./3.))
   12 CO(J*10+2)=0.0
C     SOLVE MATRIX
      CALL GELB(R,CO,400,1,2,2,5.0E-07,IER)
   40 FORMAT(5X,2HK=,I4,2X,4HIER=,I4,2X,2HI=,I4)
      WRITE(6,40)K,IER,I
      IF(K-10)51,70,70
C     SEE IF ERROR VALUES, DQ & DY, ARE LESS THAT SPECIFIED TOLERANCE
   51 DO 14 J=1,200
      ERRORQ=R(2*J)/Q(I,J)
      IF(ABS(ERRORQ)-TOL/100.)15,15,16
   15 ERRORY=R(2*J-1)/Y(I,J)
      IF(ABS(ERRORY)-TOL/100.)14,14,16
   14 CONTINUE
      GO TO 21
C     COMPUTE K+1 VALUES OF DISCHARGE AND DEPTH
   16 DO 20 J=1,200
      Q(I,J)=Q(I,J)+R(2*J)
   20 Y(I,J)=Y(I,J)+R(2*J-1)
      GO TO 17
C     COMPUTE FINAL VALUES OF DISCHARGE AND DEPTH
   21 DO 22 J=1,200
      Q(I,J)=Q(I,J)+R(2*J)
   22 Y(I,J)=Y(I,J)+R(2*J-1)
C     CALCULATE FINAL VALUES OF A, B, YB, WP, WPY, & SF FOR COLUMN I
      CALL AREA(I,RN,NC)
```

```
C   MOVE FORWARD VALUES OF A, YB & SF
       DO 18 J=1,200
       A(2,J)=A(1,J)
       YB(2,J)=YB(1,J)
    18 SF(2,J)=SF(1,J)
     2 CONTINUE
C   PRINT FINAL VALUES OF Q AND Y FOR GRID POINTS
       DO 60 J=1,200
    61 FORMAT(2X,12F10.2)
       WRITE(6,61)Q(1,J),Q(2,J),Q(3,J),Q(4,J),Q(5,J),Q(6,J),Q(7,J),Q(8,J)
      1,Q(9,J),Q(10,J),Q(11,J),Q(12,J)
    62 FORMAT(2X,12F10.4)
       WRITE(6,62)Y(1,J),Y(2,J),Y(3,J),Y(4,J),Y(5,J),Y(6,J),Y(7,J),Y(8,J)
      1,Y(9,J),Y(10,J),Y(11,J),Y(12,J)
    60 CONTINUE
    70 CONTINUE
       STOP
       END
```

```
      SUBROUTINE BEDDX
C   SUBROUTINE "BEDDX" READS IN BED SLOPE VALUES, SO, FOR EACH STATION,
C   AND, READS IN THE DISTANCE INCREMENTS, DX
      COMMON Q(12,200),Y(12,200),A(2,200),YB(2,200),SF(2,200),R(400),CO(
     11997),DX(11),SO(12),WP(200),B(200),WPY(200),YSEC(12,4),TOPW(12,4),
     2QDR(12),YDR(12)
      DO 101 L=1,12
  100 FORMAT(F10.8)
  101 READ(5,100)SO(L)
      DO 102 L=1,11
  103 FORMAT(F7.1)
  102 READ(5,103)DX(L)
  902 FORMAT(2X,'BED SLOPE (1-12) ',12F9.7)
      WRITE(6,902)SO(1),SO(2),SO(3),SO(4),SO(5),SO(6),SO(7),SO(8),SO(9),
     1SO(10),SO(11),SO(12)
  903 FORMAT(2X,'DX (1-11) ',11F9.1)
      WRITE(6,903)DX(1),DX(2),DX(3),DX(4),DX(5),DX(6),DX(7),DX(8),DX(9),
     1DX(10),DX(11)
      RETURN
      END
```

```
      SUBROUTINE DWNSTM
C   SUBROUTINE "DWNSTM" READS IN DOWNSTREAM RATING CURVE, QDR VS YDR.
C   THEN READS IN DOWNSTREAM DISCHARGE HYDROGRAPH, Q(12,J) VS TIME AND
C   COMPUTES CORRESPONDING DEPTH VALUES, Y(12,J) FROM THE RATING CURVE.
      COMMON Q(12,200),Y(12,200),A(2,200),YB(2,200),SF(2,200),R(400),CO(
     11997),DX(11),SO(12),WP(200),B(200),WPY(200),YSEC(12,4),TOPW(12,4),
     2QDR(12),YDR(12)
      DO 200 L=1,12
  201 FORMAT(F8.2)
      READ(5,201)YDR(L)
  200 READ(5,201)QDR(L)
      DO 211 J=1,200,5
  210 FORMAT(5F10.2)
  211 READ(5,210)Q(12,J),Q(12,J+1),Q(12,J+2),Q(12,J+3),Q(12,J+4)
      DO 202 J=1,200
      DO 205 L=1,12
      IF(QDR(L)-Q(12,J))205,205,206
  205 CONTINUE
  206 DIFFQ=QDR(L)-QDR(L-1)
      DIFFY=YDR(L)-YDR(L-1)
      DFQ=QDR(L)-Q(12,J)
      DFY=DFQ*DIFFY/DIFFQ
  202 Y(12,J)=YDR(L)-DFY
      RETURN
      END
```

```
      SUBROUTINE AREA(I,RN,NC)
C   SUBROUTINE "AREA" COMPUTES CURRENT VALUES OF A, B, YB, WP, WPY, & SF
      COMMON Q(12,200),Y(12,200),A(2,200),YB(2,200),SF(2,200),R(400),CO(
     11997),DX(11),SO(12),WP(200),B(200),WPY(200),YSEC(12,4),TOPW(12,4),
     2QDR(12),YDR(12)
      HTOPW2=TOPW(I,2)/2.
      HTOPW3=TOPW(I,3)/2.
      HTOPW4=TOPW(I,4)/2.
      YSEC32=YSEC(I,3)-YSEC(I,2)
      YSEC43=YSEC(I,4)-YSEC(I,3)
      WP1=2.*SQRT(YSEC(I,2)**2.+HTOPW2**2.)
      WP2=2.*SQRT(YSEC32**2.+(HTOPW3-HTOPW2)**2.)
      A1=HTOPW2*YSEC(I,2)
      A2=HTOPW2*YSEC32
      A3=HTOPW3*YSEC32
      SLOPE1=YSEC(I,2)/HTOPW2
      SLOPE2=YSEC32/(HTOPW3-HTOPW2)
      SLOPE3=YSEC43/(HTOPW4-HTOPW3)
      DO 302 J=1,200
      VY=Y(I,J)
      IF(YSEC(I,2)-VY)305,304,304
  304 HX=VY/SLOPE1
      A(NC,J)=HX*VY
      YB(NC,J)=VY/3.
      WP(J)=2.*SQRT(VY**2.+HX**2.)
      WPY(J)=2.*SQRT(SLOPE1**(-2.)+1.)
      GO TO 315
  305 IF(YSEC(I,3)-VY)307,306,306
  306 VYSEC2=VY-YSEC(I,2)
      HXT1=VYSEC2/SLOPE2
      HX=HTOPW2+HXT1
      AT2=HTOPW2*VYSEC2
```

```
      AT3=HX*VYSEC2
      A(NC,J)=A1+AT2+AT3
      WPT1=2.*SQRT(VYSEC2**2.+HXT1**2.)
      WP(J)=WP1+WPT1
      MOM1=A1*(VY-YSEC(I,2)*2./3.)
      MOMT2=AT2*(VYSEC2*2./3.)
      MOMT3=AT3*(VYSEC2/3.)
      TMOM=MOM1+MOMT2+MOMT3
      YB(NC,J)=TMOM/A(NC,J)
      WPY(J)=2.*SQRT(SLOPE2**(-2.)+1.)
      GO TO 315
  307 VYSEC3=VY-YSEC(I,3)
      HXT2=VYSEC3/SLOPE3
      HX=HTOPW3+HXT2
      AT4=HTOPW3*VYSEC3
      AT5=HX*VYSEC3
      A(NC,J)=A1+A2+A3+AT4+AT5
      WPT2=2.*SQRT(VYSEC3**2.+HXT2**2.)
      WP(J)=WP1+WP2+WPT2
      MOM1=A1*(VY-YSEC(I,2)*2./3.)
      MOM2=A2*(YSEC32*2./3.+VYSEC3)
      MOM3=A3*(YSEC32/3.+VYSEC3)
      MOMT4=AT4*(VYSEC3*2./3.)
      MOMT5=AT5*(VYSEC3/3.)
      TMOM=MOM1+MOM2+MOM3+MOMT4+MOMT5
      YB(NC,J)=TMOM/A(NC,J)
      WPY(J)=2.*SQRT(SLOPE3**(-2.)+1.)
  315 B(J)=2.*HX
  302 SF(NC,J)=RN*RN*Q(I,J)*ABS(Q(I,J))*WP(J)**(4./3.)/(2.2082*A(NC,J)**
     1(10./3.))
      RETURN
      END
```

```
C
C     ..................................................................
C
C        SUBROUTINE GELB
C
C        PURPOSE
C           TO SOLVE A SYSTEM OF SIMULTANEOUS LINEAR EQUATIONS WITH A
C           COEFFICIENT MATRIX OF BAND STRUCTURE.
C
C        USAGE
C           CALL GELB(R,A,M,N,MUD,MLD,EPS,IER)
C
C        DESCRIPTION OF PARAMETERS
C           R      - M BY N RIGHT HAND SIDE MATRIX (DESTROYED).
C                    ON RETURN R CONTAINS THE SOLUTION OF THE EQUATIONS.
C           A      - M BY M COEFFICIENT MATRIX WITH BAND STRUCTURE
C                    (DESTROYED).
C           M      - THE NUMBER OF EQUATIONS IN THE SYSTEM.
C           N      - THE NUMBER OF RIGHT HAND SIDE VECTORS.
C           MUD    - THE NUMBER OF UPPER CODIAGONALS (THAT MEANS
C                    CODIAGONALS ABOVE MAIN DIAGONAL).
C           MLD    - THE NUMBER OF LOWER CODIAGONALS (THAT MEANS
C                    CODIAGONALS BELOW MAIN DIAGONAL).
C           EPS    - AN INPUT CONSTANT WHICH IS USED AS RELATIVE
C                    TOLERANCE FOR TEST ON LOSS OF SIGNIFICANCE.
C           IER    - RESULTING ERROR PARAMETER CODED AS FOLLOWS
C                    IER=0  - NO ERROR,
C                    IER=-1 - NO RESULT BECAUSE OF WRONG INPUT PARAME-
C                             TERS M,MUD,MLD OR BECAUSE OF PIVOT ELEMENT
C                             AT ANY ELIMINATION STEP EQUAL TO 0,
C                    IER=K  - WARNING DUE TO POSSIBLE LOSS OF SIGNIFI-
C                             CANCE INDICATED AT ELIMINATION STEP K+1,
```

```
C                             WHERE PIVOT ELEMENT WAS LESS THAN OR
C                             EQUAL TO THE INTERNAL TOLERANCE EPS TIMES
C                             ABSOLUTELY GREATEST ELEMENT OF MATRIX A.
C
C        REMARKS
C           BAND MATRIX A IS ASSUMED TO BE STORED ROWWISE IN THE FIRST
C           ME SUCCESSIVE STORAGE LOCATIONS OF TOTALLY NEEDED MA
C           STORAGE LOCATIONS, WHERE
C             MA=M*MC-ML*(ML+1)/2    AND    ME=MA-MU*(MU+1)/2    WITH
C             MC=MIN(M,1+MUD+MLD),  ML=MC-1-MLD,  MU=MC-1-MUD.
C           RIGHT HAND SIDE MATRIX R IS ASSUMED TO BE STORED COLUMNWISE
C           IN N*M SUCCESSIVE STORAGE LOCATIONS. ON RETURN SOLUTION
C           MATRIX R IS STORED COLUMNWISE TOO.
C           INPUT PARAMETERS M, MUD, MLD SHOULD SATISFY THE FOLLOWING
C           RESTRICTIONS     MUD NOT LESS THAN ZERO
C                            MLD NOT LESS THAN ZERO
C                            MUD+MLD NOT GREATER THAN 2*M-2.
C           NO ACTION BESIDES ERROR MESSAGE IER=-1 TAKES PLACE IF THESE
C           RESTRICTIONS ARE NOT SATISFIED.
C           THE PROCEDURE GIVES RESULTS IF THE RESTRICTIONS ON INPUT
C           PARAMETERS ARE SATISFIED AND IF PIVOT ELEMENTS AT ALL
C           ELIMINATION STEPS ARE DIFFERENT FROM 0. HOWEVER WARNING
C           IER=K - IF GIVEN - INDICATES POSSIBLE LOSS OF SIGNIFICANCE.
C           IN CASE OF A WELL SCALED MATRIX A AND APPROPRIATE TOLERANCE
C           EPS, IER=K MAY BE INTERPRETED THAT MATRIX A HAS THE RANK K.
C           NO WARNING IS GIVEN IF MATRIX A HAS NO LOWER CODIAGONAL.
C
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C           NONE
C
C        METHOD
C           SOLUTION IS DONE BY MEANS OF GAUSS ELIMINATION WITH
```

```
C           COLUMN PIVOTING ONLY, IN ORDER TO PRESERVE BAND STRUCTURE
C           IN REMAINING COEFFICIENT MATRICES.
C
C     ..................................................................
C
      SUBROUTINE GELB(R,A,M,N,MUD,MLD,EPS,IER)
C
C
      DIMENSION R(400),A(1997)
C
C     TEST ON WRONG INPUT PARAMETERS
      IF(MLD)47,1,1
    1 IF(MUD)47,2,2
    2 MC=1+MLD+MUD
      IF(MC+1-M-M)3,3,47
C
C     PREPARE INTEGER PARAMETERS
C        MC=NUMBER OF COLUMNS IN MATRIX A
C        MU=NUMBER OF ZEROS TO BE INSERTED IN FIRST ROW OF MATRIX A
C        ML=NUMBER OF MISSING ELEMENTS IN LAST ROW OF MATRIX A
C        MR=INDEX OF LAST ROW IN MATRIX A WITH MC ELEMENTS
C        MZ=TOTAL NUMBER OF ZEROS TO BE INSERTED IN MATRIX A
C        MA=TOTAL NUMBER OF STORAGE LOCATIONS NECESSARY FOR MATRIX A
C        NM=NUMBER OF ELEMENTS IN MATRIX R
    3 IF(MC-M)5,5,4
    4 MC=M
    5 MU=MC-MUD-1
      ML=MC-MLD-1
      MR=M-ML
      MZ=(MU*(MU+1))/2
      MA=M*MC-(ML*(ML+1))/2
      NM=N*M
```

```
C
C     MOVE ELEMENTS BACKWARD AND SEARCH FOR ABSOLUTELY GREATEST ELEMENT
C     (NOT NECESSARY IN CASE OF A MATRIX WITHOUT LOWER CODIAGONALS)
      IER=0
      PIV=0.
      IF(MLD)14,14,6
    6 JJ=MA
      J=MA-MZ
      KST=J
      DO 9 K=1,KST
      TB=A(J)
      A(JJ)=TB
      TB=ABS(TB)
      IF(TB-PIV)8,8,7
    7 PIV=TB
    8 J=J-1
    9 JJ=JJ-1
C
C     INSERT ZEROS IN FIRST MU ROWS (NOT NECESSARY IN CASE MZ=0)
      IF(MZ)14,14,10
   10 JJ=1
      J=1+MZ
      IC=1+MUD
      DO 13 I=1,MU
      DO 12 K=1,MC
      A(JJ)=0.
      IF(K-IC)11,11,12
   11 A(JJ)=A(J)
      J=J+1
   12 JJ=JJ+1
   13 IC=IC+1
C
```

```
C     GENERATE TEST VALUE FOR SINGULARITY
   14 TOL=EPS*PIV
C
C
C     START DECOMPOSITION LOOP
      KST=1
      IDST=MC
      IC=MC-1
      DO 38 K=1,M
      IF(K-MR-1)16,16,15
   15 IDST=IDST-1
   16 ID=IDST
      ILR=K+MLD
      IF(ILR-M)18,18,17
   17 ILR=M
   18 II=KST
C
C     PIVOT SEARCH IN FIRST COLUMN (ROW INDEXES FROM I=K UP TO I=ILR)
      PIV=0.
      DO 22 I=K,ILR
      TB=ABS(A(II))
      IF(TB-PIV)20,20,19
   19 PIV=TB
      J=I
      JJ=II
   20 IF(I-MR)22,22,21
   21 ID=ID-1
   22 II=II+ID
C
C     TEST ON SINGULARITY
      IF(PIV)47,47,23
   23 IF(IER)26,24,26
```

```
   24 IF(PIV-TOL)25,25,26
   25 IER=K-1
   26 PIV=1./A(JJ)
C
C        PIVOT ROW REDUCTION AND ROW INTERCHANGE IN RIGHT HAND SIDE R
      ID=J-K
      DO 27 I=K,NM,M
      II=I+ID
      TB=PIV*R(II)
      R(II)=R(I)
   27 R(I)=TB
C
C        PIVOT ROW REDUCTION AND ROW INTERCHANGE IN COEFFICIENT MATRIX A
      II=KST
      J=JJ+IC
      DO 28 I=JJ,J
      TB=PIV*A(I)
      A(I)=A(II)
      A(II)=TB
   28 II=II+1
C
C        ELEMENT REDUCTION
      IF(K-ILR)29,34,34
   29 ID=KST
      II=K+1
      MU=KST+1
      MZ=KST+IC
      DO 33 I=II,ILR
C
C        IN MATRIX A
      ID=ID+MC
      JJ=I-MR-1
```

```
      IF(JJ)31,31,30
   30 ID=ID-JJ
   31 PIV=-A(ID)
      J=ID+1
      DO 32 JJ=MU,MZ
      A(J-1)=A(J)+PIV*A(JJ)
   32 J=J+1
      A(J-1)=0.
C
C     IN MATRIX R
      J=K
      DO 33 JJ=I,NM,M
      R(JJ)=R(JJ)+PIV*R(J)
   33 J=J+M
   34 KST=KST+MC
      IF(ILR-MR)36,35,35
   35 IC=IC-1
   36 ID=K-MR
      IF(ID)38,38,37
   37 KST=KST-ID
   38 CONTINUE
C     END OF DECOMPOSITION LOOP
C
C
C     BACK SUBSTITUTION
      IF(MC-1)46,46,39
   39 IC=2
      KST=MA+ML-MC+2
      II=M
      DO 45 I=2,M
      KST=KST-MC
      II=II-1
```

```
      J=II-MR
      IF(J)41,41,40
   40 KST=KST+J
   41 DO 43 J=II,NM,M
      TB=R(J)
      MZ=KST+IC-2
      ID=J
      DO 42 JJ=KST,MZ
      ID=ID+1
   42 TB=TB-A(JJ)*R(ID)
   43 R(J)=TB
      IF(IC-MC)44,45,45
   44 IC=IC+1
   45 CONTINUE
   46 RETURN
C
C
C     ERROR RETURN
   47 IER=-1
      RETURN
      END
```

The vita has been removed
from the scanned document

DEVELOPMENT OF A REVERSE FLOOD ROUTING TECHNIQUE
USING THE IMPLICIT METHOD

by

Robert N. Eli, II

(ABSTRACT)

A numerical solution technique was developed to solve the one-dimensional, partial differential equations of unsteady flow for an upstream solution. The x-t, distance-time, plane was replaced by a rectangular grid of points at which the values of the variables, discharge and depth, were computed using an implicit, finite difference scheme. A discharge hydrograph and rating curve supplied the initial values of discharge and depth along the downstream column of grid points. Boundary values of discharge were known along the top row and bottom row of grid points from steady flow conditions.

The solution proceeded from column to column in the negative x direction yielding an upstream discharge hydrograph as the final solution.

The computer model was successfully tested utilizing a reach of the James River, Virginia. The upstream hydrograph was first routed in the downstream direction using an implicit solution procedure already available.

The downstream hydrograph resulting as a solution was used as input to the reverse flood routing model. The upstream solution as computed by the model showed near perfect agreement with the actual upstream hydrograph.