STREAMFLOW ROUTING
(With Applications to North Carolina Rivers)

by

MICHAEL AMEIN AND CHING SENG FANG

Department of Civil Engineering

North Carolina State University at Raleigh

The work upon which this publication is based was supported in part by funds provided by the Office of Water Resources Research, Department of the Interior, through the Water Resources Research Institute of the University of North Carolina as authorized under the Water Resources Act of 1964.

Project No. A-001-NC
Agreement No. 14-01-0001-1401, FY 1968
January, 1969
Water Resources Research Institute
of the University of North Carolina

Report No. 17

(Revised 9/69)

ABSTRACT

In the field of water resources, it becomes frequently necessary to consider the unsteady aspects of the flow of water. Such problems arise in the planning, design, and management of water resources when computations must be made on flood flows, reservoir regulation, surges in canals, tidal flows, etc. A powerful analytical tool for the investigation of unsteady flows is available in the equations of conservation of mass and momentum. Although approximate solutions of these equations have been and still are used on a large scale, the advent of the digital computer has made it feasible to obtain complete solutions of the equations by numerical methods. In this study, three methods of solution, namely, the explicit, characteristic, and implicit methods, were investigated in detail and were applied to artificial and natural channels. The natural channel selected for this purpose was a reach of the Neuse River, North Carolina. Even though the three methods provided almost identical solutions, there is much difference among them so far as speed, reliability, simplicity, and convenience are concerned. The procedure for finding solutions by the implicit method, as developed from this study, proved to be fast, accurate, and convenient, and highly suitable for problems involving flow in natural river channels of complex geometry. This study is the first in which this type of analysis has been used. The advantages of the implicit method make it feasible to use this sophisticated tool for the solution of unsteady flow problems in a routine manner.

TABLE OF CONTENTS

Page

LIST OF TABLES iv

LIST OF FIGURES v

I. INTRODUCTION 1

II. THEORY 5

III. METHODS OF SOLUTION 11

Review of Analytical Methods 11
Numerical Methods 13

Approximate Numerical Methods 13
Complete Numerical Methods 15

IV. METHOD OF CHARACTERISTICS 18

Numerical Procedure 20

Initial Characteristic 20
Interior Characteristics 20
Upstream Boundary 22
Downstream Boundary 23

Solution Technique 24

V. EXPLICIT METHOD 27

Numerical Procedure 27

Interior Points 27
Upstream Boundary 29
Downstream Boundary 31

VI. IMPLICIT METHOD 33

Numerical Procedure 33
Generalized Newton Iteration Method 40
Solution of the Finite Difference Equations . . 42

VII. APPLICATIONS 46

Computer Programs 47
Unsteady Flow Through an Idealized Channel . . 47

TABLE OF CONTENTS (continued)

Page

Approximate Method 48
Explicit Method 48
Characteristic Method 49
Implicit Method 50
Comparison of Results and Conclusions 51

Unsteady Flow Through a Natural Channel 53

Initial and Boundary Data 55
Computations for the Composite Section Channel 57
Computations for the Discrete Section Channel Model 62
Results of Computations 62
Computation Times 66

Discussion of Results 68

VIII. SUMMARY AND CONCLUSIONS 69

REFERENCES 71

APPENDIX 73

LIST OF TABLES

Page

1. Polynomial coefficients of equations for individual cross-sectional areas 63

2. Computer times on an IBM System 360/75 for calculating flood flows 66

LIST OF FIGURES

		Page
1.	Definition sketch for open channel flow	6
2.	Network of characteristics on the (x,t) plane	21
3.	Network of points on the (x,t) plane for the explicit method	28
4.	Network of points on the (x,t) plane for computing on the upstream boundary by the explicit method	30
5.	Network of points on the (x,t) plane for computations at the downstream boundary	32
6.	Network of points on the (x,t) plane for the implicit method	34
7.	Stage hydrographs by the explicit method	49
8.	Stage hydrographs by the method of characteristics	50
9.	Stage hydrographs by the implicit method	51
10.	Comparison of hydrographs by three complete methods	52
11.	Neuse River between Goldsboro and Kinston, North Carolina	53
12.	Measured flood hydrographs at Goldsboro, North Carolina	54
13.	Rating curves at Goldsboro and Kinston, North Carolina, in 1945	56
14.	Composite section area and top width	57
15.	Manning's roughness coefficient at Goldsboro and Kinston, North Carolina, for the flood in 1945	59
16.	Calculated 1945 flood hydrograph near Kinston, North Carolina (composite section)	60
17.	Calculated 1964 flood hydrograph near Kinston, North Carolina (composite section)	61
18.	Typical channel sections taken from small-scale topographic maps	64

LIST OF FIGURES (continued)

Page

19. Calculated 1945 flood hydrograph near Kinston, North Carolina, by the implicit method (discrete section) 66

20. Calculated 1964 flood hydrograph near Kinston, North Carolina, by the implicit method (discrete section) 67

I. INTRODUCTION

A fascinating but at the same time very challenging characteristic of problems in water resources is the unsteady nature of the phenomena involved. The unsteadiness or the change with time may be relatively slow, such as the loss due to evaporation from a standing body of water or the gradual contamination of a lake over a number of years. The change also may be rapid, such as the movement of a water wave hurling down a channel after the failure of a dam. Hardly a problem exists in the water resources field for which conditions are not unsteady, even though for the sake of simplifying the analysis steady-state assumptions are often made.

This report summarizes the results of an investigation on the unsteady flow in streams, which is a major problem in water resources. The investigation is based on the applications of the laws of mechanics, which are known to be the most sophisticated and powerful analytical tool for this type of study. It is recognized, however, that this is not the only approach to the study of this problem, and other approaches may have many commendable features.

Streamflow, together with overland flow, comprises the surface runoff component of the hydrological cycle. Overland flow is the runoff from land areas such that flow takes place at shallow depths and the water does not travel in well-defined channels. The rate of change of runoff in the direction of travel, known as lateral inflow, is the most important feature of overland flow. Streamflow is the flow in well-defined channels and is composed of the accumulation of overland flows. The changes in flow rates due to lateral inflows play a secondary role in streamflow. The primary role is played by the flow in the main channel, and significant changes in flow occur at confluences with tributaries.

The physical laws governing the flow of water in overland flow and in channel flow are the same, whether the flow is from a parking lot, a tract of land, or a large river basin. The same techniques can be effective, at least in principle, in the study of both phenomena. However, the development of surface runoff hydrographs for complex watersheds on the basis of laws of mechanics is beset with great uncertainties. We can hope to be far more successful where the problem involves the flow over a simple topography, such as a paved parking lot. Application of the laws of mechanics becomes a dubious enterprise if we have to follow the movement of water through the woods, fields, farms, hills, and valleys. The development of watershed hydrographs will, therefore, of necessity require the use of much empiricism in the foreseeable future.

Application of the laws of mechanics to streamflow is much more promising because the flow is taking place in well-defined channels. Streamflow covers a wide range of magnitudes, from the flow in small roadside drainage ditches to the flow in mighty rivers. Although the physical principles governing flow in channels are the same regardless of the size of the channel and the magnitude of the flow rate, different mathematical procedures may be required for different cases.

Well-established procedures, designated as backwater and drawdown computations, for the determination of the steady flow properties such as average velocity and water depth are known. However, methods for determining the same quantities are far from being well established for unsteady flows, and the determination of the velocities and depths under unsteady conditions is a problem of far greater complexity. If we focus our attention on the movement of a flood wave advancing into a long river channel, the flood wave may change in form, it may subside, or it may become peaked at the crest. Under some conditions, it may steepen at the front and develop a bore. Although all flows in nature exhibit varying degrees of unsteadiness, the movement of floods is a prime example of unsteady flow. Furthermore, it is of vital importance to the life and works of man. It is imperative to know to what depths and how frequently the flood plains on which man has built will be flooded, at what times, and to what heights will the flood waters rise so that adequate measures for evacuation and salvaging of property could be made, and to what capacities reservoirs for impounding flood waters should be constructed so that water levels along the river are maintained below dangerous levels. We also may be interested in establishing procedures for regulation of reservoirs. The answers to these and many other questions could be supplied by the techniques of flow routing. These techniques determine the water depths and velocities in channels, rivers, and reservoirs under unsteady conditions. The techniques are commonly applied to flood motions and are consequently commonly known as flood routing.

The laws of mechanics on which flood-routing techniques are based consist of the laws for the conservation of mass, momentum, and energy. Since in river flows thermodyamic effects can often be neglected, the laws of conservation of momentum and energy become nearly identical. The laws for the conservation of momentum and mass are expressed in the form of mathematical equations known as the equations of unsteady flow. Application of flood-routing techniques consists of finding solutions to these equations.

Streamflow routing methods may be classified into two methods--approximate and complete. In the approximate methods,

the equations of motion are simplified; in the complete methods, no simplifications are made. Although the approximate methods provide correct solutions to some problems, they will be found unreliable in other cases. Analytical solutions of complete equations are not feasible; therefore, all complete methods for the solutions of the equations of unsteady flow are numerical.

The advent of the digital computer has made it feasible to obtain the numerical solutions of the complete equations, and much effort has gone into this in recent years. This report is an outcome of such an effort. A major objective of this investigation is to make the solutions accessible to most workers in the field of water resources. It is hoped that once practical computer programs are prepared, the user could find solutions to problems without being required to possess a knowledge of the mathematical details which have gone into their preparation.

In this report, we treat the application of the equations of unsteady flow to flood routing only. It should be recognized that most of the methods and principles applied to flood routing also are applicable, with slight modifications, to other problems involving the unsteady motion of water, such as the movement of surges in canals, tidal motion, long waves, etc. We believe, however, that flood routing is the most challenging of all problems involving the equations of motion. In flood routing, both the friction force and the inertia force play important roles, and we cannot ignore one with respect to the other as could be done, for instance, in the case of a canal surge. Again, flood flows, unlike most other unsteady flows, may have long durations, say of several days. It becomes a very challenging matter to find realistic solutions to flood flow problems in a reasonable time. Numerical methods that may be very practical for most unsteady flow problems will fail when they are applied to flood routing, because it takes too long to deliver the answers. The study of flood routing not only sheds light on a very important phenomenon in water resources, it also is aimed at a most challenging area for the application of the equations of unsteady flow.

This report contains a detailed treatment of three numerical methods for the solution of the equations of unsteady flow and of their applications. The three methods represent unique approaches to the solution of the equations of unsteady flow. The methods are applied to two types of problems, viz., flow through an idealized channel and flow through a natural channel. The objective of the first application is to illustrate the mathematical aspects of the methods and to explore basic principles. The objective of the second type of application is to investigate the feasibility of treating real problems.

This report is not intended to contain the final word on numerical flood routing. Its main concern is the comparison of various numerical routing methods, the study of causes for lack of general acceptance of some of the methods, and the search for methods that could be efficient and workable. The report also is incomplete in the sense that we did not have the time and resources available to test the results on a variety of problems. Because of the magnitude of the effort involved, we had to confine ourselves to a single natural channel. We believe, however, that this work and the work of other investigators deeply involved in this type of effort will have laid the groundwork for the application of numerical flood routing to water resources in a routine manner.

II. THEORY

The basic laws of mechanics which serve as the basis for the investigation of problems of surface runoff and other phenomena involving the movement of water are the laws for the conservation of mass and momentum. These laws are expressed in the form of mathematical equations known as the equations of unsteady flow. They also are referred to as St. Venant's equations, after one of the pioneering theoreticians. The derivation of the equations of unsteady flow is given in standard reference works (e.g., Chow, 1959; Gilcrest, 1950; Henderson, 1966; Stoker, 1957). Their derivation will be given here with some minor deviation from that in some of the cited references. The argument for our approach will be introduced in the course of the presentation.

Consider a short reach of channel of length Δx with the flow taking place from section 1-1 to section 2-2 as shown in Fig. 1. Let x be the horizontal distance; ρ, the density of water; g, the acceleration due to gravity; and let z denote the channel bottom elevation; y, the water depth, A, the cross-sectional area; and v, the average velocity. Let Q be the volume flow rate entering the channel through section 1-1; $Q + (\partial Q/\partial x)\Delta x$, the volume outflow rate; and let q be the lateral inflow per unit channel length per unit time. The mass of water entering the channel reach during a time interval Δt will be $\rho Q \Delta t + \rho q \Delta x \Delta t$.

The mass of water leaving the channel during the same time would be

$$\rho \left(Q + \frac{\partial Q}{\partial x} \Delta x\right)\Delta t.$$

Assuming $\partial Q/\partial x$ to be positive, there is a net mass outflow from the reach. The law of conservation of mass would require that because of the net mass outflow during time Δt the mass inside the reach be reduced and consequently the water surface should fall. The mass of water inside the reach of length Δx is

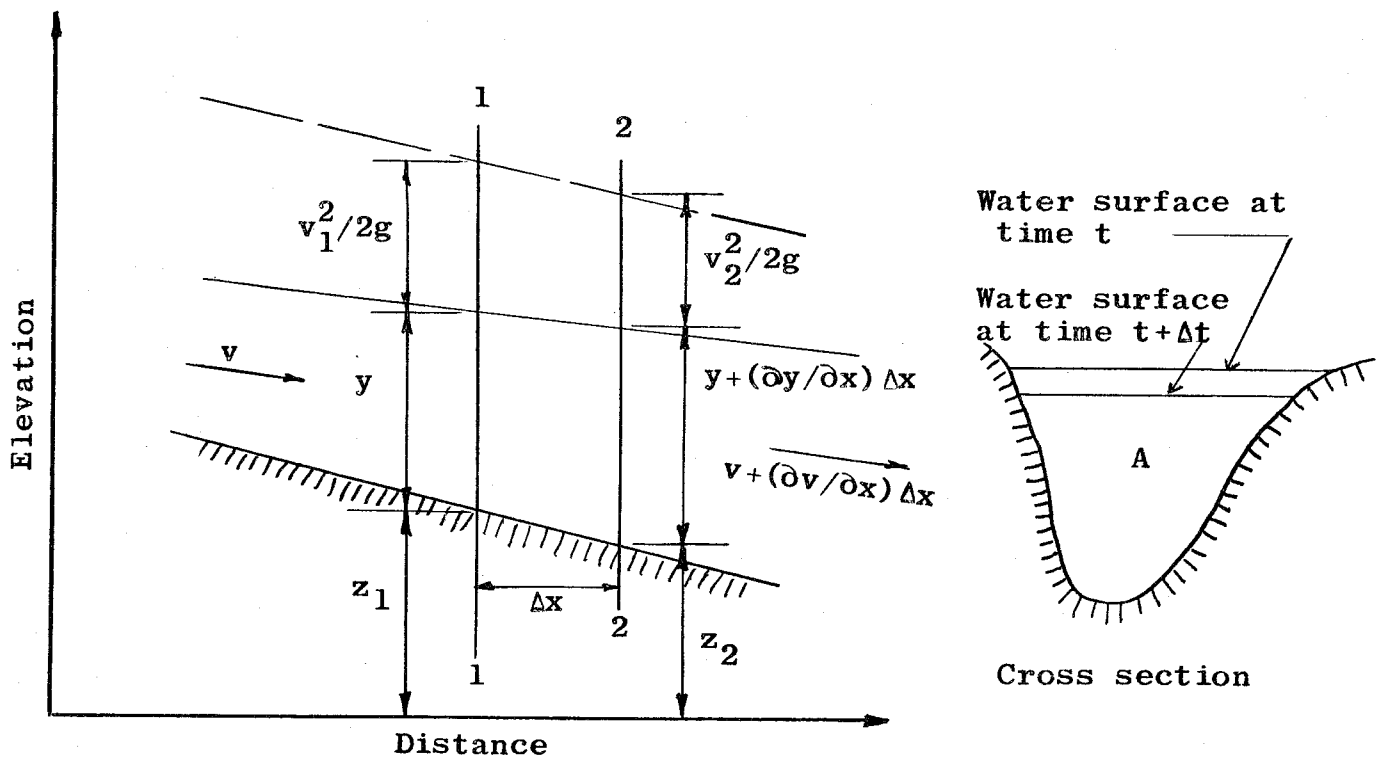
$$\rho A \ \Delta x.$$

The rate of decrease of this mass can be expressed as

$$- \frac{\partial}{\partial t}(\rho A \Delta x) = -\rho \Delta x \frac{\partial A}{\partial t}.$$

The reduction in mass during time interval Δt is

$$-\rho \Delta x \left(\frac{\partial A}{\partial t}\right) \cdot \Delta t.$$



Longitudinal section

Fig. 1. Definition sketch for open channel flow

For the conservation of mass we equate the net mass outflow to the reduction in mass inside the reach, so that

$$\rho(Q + \frac{\partial Q}{\partial x}\Delta x)\Delta t - (\rho Q\Delta t + \rho q\Delta x\Delta t) = -\rho \frac{\partial A}{\partial t}\Delta t\cdot\Delta x.$$

Simplifying,

$$\frac{\partial Q}{\partial x} = -\frac{\partial A}{\partial t} - q \quad \text{or} \quad \frac{\partial Q}{\partial x} + \frac{\partial A}{\partial t} + q = 0. \tag{1}$$

Equation (1) is known as the equation of continuity and is the mathematical expression for the law of conservation of mass. Since $Q = vA$, the equation of continuity may be written as

$$v\frac{\partial A}{\partial x} + A\frac{\partial v}{\partial x} + \frac{\partial A}{\partial t} + q = 0. \tag{1a}$$

The conservation of momentum is given by Newton's second law of motion which states that the rate of change of momentum is equal to the applied force. Let us consider the same element of volume shown in Fig. 1. The applied force on this element is the resultant of the pressure, gravity, and friction forces on the element. The water depth is y at section 1-1 and it is $y + (\partial y/\partial x)\Delta x$ at section 2-2. The cross-sectional area is A at 1-1 and is $A + (\partial A/\partial x)\Delta x$ at 2-2.

The pressure distribution is assumed to be hydrostatic. This assumption is valid if the surface curvature is small, and is true of most flows in open channels. The pressure force at section 1-1 acts toward the right, and the pressure at section 2-2 acts toward the left. As it is assumed in the figure that the depth at section 2-2 is greater than at section 1-1, there is an unbalanced pressure at section 2-2 equal to $\rho g(\partial y/\partial x)\Delta x$. The unbalanced pressure force is, therefore,

$$\rho g A \frac{\partial y}{\partial x}\Delta x$$

and is directed toward the negative x-axis.

The gravity force is equal to the weight of the fluid inside the element, and is

$$\rho g\ A\ \Delta x.$$

The component of the gravity force in the direction of motion is

$$-\rho g A\Delta x \frac{\Delta z}{\Delta x}.$$

The channel bottom slope $\Delta z/\Delta x$ is expressed as S_o. Therefore, the gravity force in the direction of motion is

$$\rho g A S_o \Delta x.$$

The friction force opposes the motion and is directed to the left. If the shear stress is designated as τ, the shear force becomes

$$\tau P \Delta x,$$

where P is the wetted perimeter.

The shear force can be expressed in terms of the head (or energy) loss by equating the head loss to the work done by the shear force. Thus,

$$\tau P \Delta x = \gamma h_L A,$$

where h_L is the head loss over distance Δx. The head loss itself can be expressed in terms of the friction slope, S_f, so that

$$h_L = S_f \Delta x.$$

Therefore, the friction force is equal to

$$\tau P \Delta x = \gamma S_f A \Delta x.$$

The resultant force on the element of volume in the direction of motion is the resultant of the gravitational, frictional, and pressure forces:

$$\rho g A S_o \Delta x - \rho g A S_f \Delta x - \rho g A \frac{\partial y}{\partial x} \Delta x.$$

The rate of change of momentum of the fluid through the element is determined as follows. The change in momentum consists of two parts. The first part is the change with time in the momentum inside the volume element. This change is termed the local momentum change. The second part of the momentum change is due to the difference in the momentum entering and leaving the volume elements. This rate of momentum flow is termed the convective change in momentum. The local change is a temporal change, while the convective change is a spatial change.

The momentum of the fluid inside the volume element is $(\rho A \Delta x)v$. Therefore, the local momentum change is

$$\frac{\partial}{\partial t}(\rho A \Delta x)v = \rho \Delta x \left(A \frac{\partial v}{\partial t} + v \frac{\partial A}{\partial t} \right).$$

The momentum inflow rate into the volume element is

$$\rho vA \cdot v = \rho v^2 A.$$

The convective (spatial) momentum change is, therefore,

$$\frac{\partial}{\partial x}(\rho v^2 A)\Delta x = \rho v^2 \frac{\partial A}{\partial x}\Delta x + 2\rho v \frac{\partial v}{\partial x} A\Delta x$$

$$= \rho v \Delta x (v \frac{\partial A}{\partial x} + 2A \frac{\partial v}{\partial x}).$$

The total rate of change of momentum is the sum of the local and convective momentum changes; i.e.,

$$\rho \Delta x \left(\frac{A\partial v}{\partial t} + \frac{v\partial A}{\partial t}\right) + \rho v \Delta x \left(\frac{v\partial A}{\partial x} + \frac{2A\partial v}{\partial x}\right).$$

But, from the equation of continuity (1),

$$\frac{A\partial v}{\partial x} + \frac{v\partial A}{\partial x} = -q - \frac{\partial A}{\partial t}.$$

Substituting this relationship in the expression for momentum, the rate of change of momentum becomes

$$\rho A \Delta x \left(\frac{\partial v}{\partial t} + \frac{v\partial v}{\partial x}\right) - \rho v q \Delta x.$$

Finally, the application of Newton's second law to the flow through the element of volume becomes

$$\rho g A S_o \Delta x - \rho g A S_f - \rho g A \frac{\partial y}{\partial x}\Delta x = \rho A \Delta x \left(\frac{\partial v}{\partial t} + \frac{v\partial v}{\partial x}\right) - qv\Delta x.$$

Dividing all the terms by $\rho A \Delta x$, the equation simplifies to

$$gS_o - gS_f - g\frac{\partial y}{\partial x} = \frac{\partial v}{\partial t} + v\frac{\partial v}{\partial x} - \frac{qv}{A}$$

or

$$\frac{\partial v}{\partial t} + v\frac{\partial v}{\partial x} = gS_o - g\frac{\partial y}{\partial x} - gS_f - \frac{qv}{A}. \quad (2)$$

Equation (2) is the mathematical expression for the conservation of momentum.

A simpler but less rigorous approach to the application of Newton's second law is to express it as

Force = Mass x acceleration,

and apply it to the flow through an element of volume. Referring again to Fig. 1, the mass of the fluid inside the element would be $\rho A \cdot \Delta x$. The acceleration of each fluid unit mass would be

$$\frac{dv}{dt} = \frac{\partial v}{\partial t} + \frac{v\partial v}{\partial x},$$

where the first term in this equation is the local (temporal) acceleration and the second term is the convective (spatial) acceleration. Therefore, the product of mass and acceleration becomes

$$\rho A \cdot \Delta x \cdot \left(\frac{\partial v}{\partial t} + \frac{v\partial v}{\partial x}\right).$$

The applied force is the resultant of the pressure, gravity, and friction forces and has been found already to be

$$\rho g A S_o \cdot \Delta x - \rho g A S_f \cdot \Delta x - \rho g A \frac{\partial y}{\partial x} \Delta x.$$

Equating the applied force to the product of mass and acceleration, and dividing all the terms by $\rho g A \cdot \Delta x$, we again end up with equation (2).

It should be noted that in equation (2) the lateral inflow on entering the channel is assumed to possess no momentum in the direction of flow.

Equations (1) and (2) are the equations of unsteady flow. They serve as the basis for constructing mathematical models for the investigation of numerous problems. Application of the equations to the unsteady motion of water, whether it be the propagation of surges in canals, overland flow, or the progress of a flood in a river channel, consists of finding the solutions of the equations in conformance with the boundary conditions. The boundary conditions may consist of the known changes in water levels at two ends of a reach, in which case the solutions determine the velocities and water depths at intermediate points as functions of time. For most flood flow problems, the boundary conditions are given as a stage or discharge hydrograph at the upstream end of a reach and the properties of some control section, such as a dam at the downstream end. The objective of finding the solutions is to determine the water depths and velocities at all points in the reach as functions of time.

III. METHODS OF SOLUTION

From a mathematical point of view, the equations of unsteady flow are classified as nonlinear partial differential equations of the hyperbolic type. Two approaches to finding solutions of the equations have been used. These are: analytical methods and numerical methods. In the following paragraphs, a brief review and discussion of these methods are presented.

Review of Analytical Methods

The equations of unsteady flow are not amenable to general analytical solutions. Any analytical solutions that might be attempted must of necessity be based on first reducing the equations to simpler forms. In this procedure, some important features of the equations could be lost, and the resulting solutions might lack generality. The results obtained in this manner must be used with great care, and it should be recognized that although the solutions might be valid for some applications they might fail in other cases.

It is not intended here to present a comprehensive review of the available solutions of the equations of unsteady flow. A brief review of representative analytical solutions is given to provide the proper background for the central theme of this report.

A simple concept for the flood wave is given by what is termed the monoclinical progressive wave. This wave, which may actually occur in nature, is assumed to travel with a constant shape and a constant speed. These two assumptions make it feasible to study the movement of the wave as a steady flow phenomenon by referring the motion to a moving coordinate system. An observer traveling at the wave speed in the same direction as the wave would see a constant profile. Application of the equation of continuity leads to the following equation for wave speed (Gilcrest, 1950):

$$v_w = \frac{dQ}{dy},$$

where Q is the discharge and y is the depth.

More recently Lighthill and Whitham (1955) introduced the concept of the kinematic wave. The kinematic wave is defined as one in which the discharge is a function of depth only. In the kinematic wave theory, the equation of continuity is used by itself and the momentum equation is drastically simplified.

Another approach to finding analytical solutions of equations (1) and (2) is the use of diffusion analogy. The main concepts can be better illustrated by confining our attention to the one-dimensional case. For this case, equations (1) and (2) reduce to

$$\frac{\partial y}{\partial t} + \frac{\partial}{\partial x} vy = 0$$

$$\frac{\partial v}{\partial t} + v \frac{\partial v}{\partial x} + g \frac{\partial y}{\partial x} - gS_o - gS_f = 0.$$

In diffusion analogy, the acceleration terms are neglected so that the dynamic (momentum) equation reduces to

$$g \frac{\partial y}{\partial x} = gS_o - gS_f \quad \text{or} \quad S_f = S_o - \frac{\partial y}{\partial x}.$$

The friction slope S_f may be given by one of the well-known resistance formulas. Using Chezy's formula,

$$S_f = v^2/C^2y,$$

where C is Chezy's coefficient and

$$v = C\sqrt{yS_f} = C\sqrt{y}\ \sqrt{S_o - (\partial y/\partial x)}.$$

Substituting in the equation of continuity, we find

$$\frac{\partial y}{\partial t} + \frac{\partial}{\partial x}\left[Cy^{3/2}\ \sqrt{S_o - (\partial y/\partial x)}\right] = 0$$

or

$$\frac{\partial y}{\partial t} + \frac{3}{2}C\sqrt{y}\ \frac{\partial y}{\partial x}\ \sqrt{S_o - (\partial y/\partial x)} - \frac{1}{2}Cy^{3/2}\left(S_o - \frac{\partial y}{\partial x}\right)^{-1/2} \frac{\partial^2 y}{\partial x^2} = 0.$$

This equation may be approximated to

$$\frac{\partial y}{\partial t} + \frac{3}{2}C\sqrt{yS_o}\ \frac{\partial y}{\partial x} = \frac{1}{2}C(y^{3/2}/\sqrt{S_o})\frac{\partial^2 y}{\partial x^2}.$$

and has the same form as the diffusion equation. If the coefficients of $\partial y/\partial x$ and $\partial^2 y/\partial x^2$ are assumed to be constant, the equation reduces to the classic linear diffusion equation (Chow, 1959), i.e.,

$$\frac{\partial y}{\partial t} + k \frac{\partial y}{\partial x} = N \frac{\partial^2 y}{\partial x^2},$$

which has been extensively applied in heat transfer and other areas. We observe that in order to reduce the equations of

unsteady flow to the classic diffusion equation considerable approximation is necessary. The results obtained from diffusion analogy would be valid for applications in which the assumptions are justified.

All attempts at finding analytical solutions of the equations of unsteady flow, as in the principal methods described above, are based on simplifying assumptions. Although the attempts may involve highly sophisticated analyses, we cannot depend on these methods to determine actual flood flow movements except in special cases.

Numerical Methods

Numerical methods for the solution of differential equations are established on the basis of replacing the derivatives by finite differences. The differential equations are then represented by corresponding algebraic equations. Although numerical methods do not require that any drastic simplifications be introduced into the equations of unsteady flow, the procedure for representing the partial differential equations by algebraic equations is not straightforward. Many factors involving convergence, stability, accuracy, and efficiency of the numerical procedures enter the picture. Although the procedure involved may be reduced to simple mathematical operations, the number of those operations may be extremely large, and is particularly true for application to most flood-flow problems. Therefore, until recently, numerical methods for the solution of the equations of unsteady flow have consisted mostly of approximate methods, in which the equations were simplified to reduce the number of computational steps. The advent of the digital computer made it feasible to attempt numerical solutions of the complete (unsimplified) equations. Because of long computation times and storage requirements, even with digital computers, much work has gone into the development of approximate methods. The review of numerical methods, therefore, includes both approximate and complete methods.

Approximate Numerical Methods

Nearly all approximate numerical methods are based primarily on the equation of continuity. The dynamic equation is either neglected or is drastically simplifed. In some cases, empirical techniques are substituted for the dynamic equation.

The best known approximate methods for the integration of the equations of unsteady flow are the so-called storage routing methods. In a typical method, the equation of

continuity (neglecting lateral inflow) is represented in finite difference form as

$$\frac{\Delta Q}{\Delta x} = \frac{\Delta A}{\Delta t} \quad \text{or} \quad \Delta Q = \frac{\Delta A}{\Delta t}\,\Delta x.$$

Since $A\Delta x$ is the volume of water in the subreach Δx, $\Delta A \Delta x$ is the change in the volume or the change in storage and can be represented as

$$\Delta S = \Delta A \cdot \Delta x,$$

where S is the storage. The equation of continuity reduces, therefore, to

$$\Delta Q = \Delta S/\Delta t.$$

But $\Delta Q = O - I$, where O is the outflow rate and I is the inflow rate. Thus the equation of continuity can be represented as

$$O - I = \Delta S/\Delta t.$$

This algebraic equation is used to route flows through reservoirs or through subreaches of rivers.

To carry out computations it is necessary that a relationship between the storage and outflow be known. For most reservoirs, a consistent relationship may be determined. For river channels, the storage-outflow relationship may not be stable. Consequently, various relationships, the most common of which is one based on deriving storage-discharge relationship assuming steady flow, have been proposed. Although this storage-routing method provides realistic values for outflow from most reservoirs, its success in predicting unsteady flows through river channels is a subject of considerable controversy. According to Thomas (1934) solutions are heavily dependent on the choice of values for Δx and Δt, and one could possibly arrive at any desired solution by selecting appropriate values for these parameters. Although the reliability of the storage routing methods can be questioned, they are traditionally popular in engineering practice because of their simplicity and also because they are manageable by desk calculators.

Another widely used storage-routing method is the Muskingum method, named after the Muskingum River, Ohio, to which it was first applied. In this method the storage-discharge relationship is given by an empirical equation, the coefficients of which are determined from historical records. Details of the storage routing methods mentioned here and of other storage routing methods are given in most standard reference works (e.g., Linsley et al., 1949; Gilcrest, 1959; Chow, 1960).

A somewhat different approach toward the establishment of approximate numerical methods is illustrated by Brakensiek and Comer (1966), where the equation of continuity is retained but the momentum equation is simplified to

$$\frac{\partial y}{\partial x} = S_o - S_f$$

by neglecting the acceleration terms. The equation of continuity and the simplified momentum equation constitute a system of two first-order partial differential equations. Expressing the friction slope S_f as $S_f = (Q/K)^2$, where K is the conveyance, this system can be reduced by differentiation to the following single, second-order differential equation:

$$\frac{\partial Q}{\partial t} = \frac{K^2}{2QB}\frac{\partial^2 Q}{\partial x^2} - \left(\frac{Q}{BK}\frac{dK}{dy}\right)\frac{\partial Q}{\partial x} + \left(\frac{q}{BK}\frac{dK}{dy}\right)Q,$$

where B is the channel top width. This is a nonlinear partial differential equation of the parabolic type and is a diffusion analogy. The simplifications introduced in this approach make the solutions manageable for computers with small storage capacity.

Complete Numerical Methods

Approximate methods, whether analytical or numerical, provide solutions which are useful, accurate, and relatively easy to obtain for a great variety of problems. However, the user is almost always in doubt as to the extent to which the accuracy of the solution is affected by the approximations. In some cases, the approximation might be the cause of glaring discrepancies between the true and approximate solutions, while in other cases such differences may not be noticeable. Development of methods for the solution of the complete equations has been the concern of many investigators, and procedures for their numerical solution have been proposed from time to time. But they could not be verified because of the exorbitant amount of tedious work required to carry out the computations. Serious attempts at finding solutions of the complete equations could be made with high-speed computers. Accompanying the advances in computing machinery, great strides have also been taken in the science of numerical methods and in the theory of partial differential equations, so that numerical procedures can be established on sound theoretical principles.

For the purposes of this report, methods for the numerical solution of the complete equations of unsteady flow

based on finite difference techniques are classified as direct methods and characteristic methods. We define direct methods as those in which the finite difference representation is based directly on the primary equations. In the characteristic methods, the equations are first transformed into the characteristic form and the latter form is used in finite difference representation. A fixed mesh of points on the time-distance plane is commonly employed to identify points at which solutions are obtained by the direct methods. In the characteristic methods, solutions may be obtained at the intersection of characteristic curves on the time-distance plane or at fixed points of a rectangular mesh by interpolation between the intersection of characteristics.

The finite difference schemes used in the direct and the characteristic methods are further classified into explicit and implicit methods. In the explicit methods, the finite difference equations are usually linear algebraic equations from which the unknowns can be evaluated explicitly a few at a time. The unknowns occur implicitly in the finite difference equations of implicit methods, which are usually nonlinear algebraic equations.

On the basis of the above classification, the following methods for the solution of the equations of unsteady flow can be used:

1. Explicit characteristic using characteristic network,
2. Implicit characteristic using characteristic network,
3. Explicit characteristic using fixed mesh,
4. Implicit characteristic using fixed mesh,
5. Direct explicit using fixed mesh, and
6. Direct implicit using fixed mesh.

Each method could have many variations depending on what type of mesh is selected and what procedure for the solution of the finite equations is used.

This report contains a detailed treatment of three numerical methods for the solution of the equations of unsteady flow and of their applications. The three methods are:

1. The implicit characteristic method using characteristic network,
2. The explicit direct method, and
3. The implicit direct method.

The other three methods cited in the preceding paragraph were not investigated as part of this study because they were found to be either unstable for general flood routing problems (Amein, 1966) or they were basically similar to one of the methods investigated. The three methods presented in this report represent unique approaches to the solution of the equations of unsteady flow. Although the explicit direct method has been thoroughly studied (Stoker, 1957), and this report contains little innovation on its application, it serves as a reference against which the efficiency and usefulness of other methods of solution can be measured.

For the sake of convenience in the following pages of this report, the implicit characteristic method using the characteristic network will be referred to as the characteristic method. The explicit direct method and the implicit direct method will be designated as the explicit and implicit methods, respectively. This abbreviated nomenclature focuses on the most distinguishing feature of each method and identifies it. In this report, therefore, the characteristic method employs a curvilinear network of characteristics, while the explicit and implicit methods utilize fixed meshes. The three methods are applied to two types of problems, viz., (1) flow through an idealized channel and (2) flow through a natural channel. The objective of the first application is to illustrate the mathematical aspects of the methods and to explore basic principles. This report will touch on these aspects rather briefly because they have been discussed in detail in several recent papers (Amein, 1966, 1967, 1968). The objective of the second type of application is to investigate the feasibility of treating real problems and, if practicable, to provide a useful tool for workers in water resources.

IV. METHOD OF CHARACTERISTICS

The equations of unsteady flow that were derived in Section II are reproduced below for convenience.

$$\frac{\partial A}{\partial t} + A\frac{\partial v}{\partial x} + v\frac{\partial A}{\partial x} - q = 0 \tag{1}$$

$$\frac{\partial v}{\partial t} + v\frac{\partial v}{\partial x} = gS_o - g\frac{\partial y}{\partial x} - gS_f - \frac{qv}{A}. \tag{2}$$

All the terms retain their definitions from Section II.

In the method of characteristics, these two partial differential equations are transformed into a system of four ordinary differential equations. The theory of the method of characteristics was given by Courant and Friedrichs (1948), Crandall (1957), and others. Recent applications of the method to gradually varied unsteady flows employing a characteristic network were reported by Amein (1966), Liggett and Woolhiser (1967), and Fletcher and Hamilton (1967). Baltzer and Lai (1968) used a fixed-mesh characteristic method for the study of unsteady flows through tidal reaches and estuaries.

We derive the equations of characteristics following Courant and Friedrichs (1948). The geometry of the channel cross section is a function of depth; therefore,

$$A = A(y),\ B = B(y),\ R = R(y),$$

where A, B, and R are, respectively, the cross-sectional area, the top width, and the hydraulic radius. Since in unsteady flow the depth may change with distance and time, we can write

$$\frac{\partial A}{\partial t} = \frac{dA}{dy}\frac{\partial y}{\partial t},\ \frac{\partial A}{\partial x} = \frac{dA}{dy}\frac{\partial y}{\partial x},\ \ldots.$$

It should be noted that dA/dy is the channel width B(y) at the depth y. If the values of A(y) and B(y) are obtained by independent measurements, the measurement errors may cause B(y) to be different from the values obtained by differentiating the area A(y) with respect to depth. For numerical stability it is important that A(y) and B(y) should be compatible. Therefore, if either A(y) or B(y) is obtained by measurement, the other should be determined by calculus. The equations of unsteady flow may be written as

$$A\frac{\partial v}{\partial x} + v\frac{dA}{dy}\frac{\partial y}{\partial x} + \frac{dA}{dy}\frac{\partial y}{\partial t} - q = 0$$

$$v\frac{\partial v}{\partial x} + \frac{\partial v}{\partial t} + g\frac{\partial y}{\partial x} + \frac{qv}{A} - gS_o + gS_f = 0.$$

The slopes of the characteristics are given by (Courant and Friedrichs, 1948):

$$\frac{dx}{dt} = u \pm c, \qquad (3) \text{ and } (4)$$

where $c = \sqrt{gA/(dA/dy)}$. The positive sign applies to the advancing (or forward) characteristic while the negative sign applies to the receding (backward) characteristic. Furthermore, the equation,

$$dv + \sqrt{g(dA/dy)/A}\, dy - g(S_o - S_f - \frac{qv}{A})dt - qdx = 0, \qquad (5)$$

holds on the advancing characteristic and the equation,

$$dv - \sqrt{g(dA/dy)/A}\, dy - g(S_o - S_f - \frac{qv}{A})dt - qdx = 0, \qquad (6)$$

holds on the receding characteristic. Equations (3) through (6) are four ordinary differential equations replacing the partial differential equations (1) and (2). These four equations serve as the basis for numerical solution of equations of unsteady flow by the method of characteristics.

We may prescribe the initial and boundary conditions that would be necessary for the complete solution of the equations in several ways. Let us consider a typical flood flow through a channel reach. For the sake of convenience, let $x = 0$ at the upstream end and let $t = t_o$ at the start of computations. The required initial values are the values of v and y at time $t = t_o$ at all points in the reach; that is, for all x. The boundary conditions may be given as

1. The values of y or of Q at $x = 0$ as functions of time.

2. A relationship between v and y at the downstream end.

Boundary conditions prescribed in this way are typical of most flood-flow situations in subcritical flow. Thus the upstream boundary conditions may be given by a stage or discharge hydrograph and the downstream boundary conditions by a rating curve or the properties of the control section. It is appropriate to utilize a control section such as a dam or weir as the downstream boundary for a reach. Other types of problems may require different initial and boundary conditions. For a tidal inlet, for example, the stage variations with time at the two ends of the inlet channel may be known and it might be desired to find the changes in water levels at intermediate locations in the inlet and the flow velocities at all points of the inlet as functions of time.

The following paragraphs contain the details of the numerical procedure for a typical flood-routing problem. The objective of the procedure is to find the solutions of the equations of unsteady flow as the values of flow variables at all points of intersection on a curvilinear network of characteristics on the x,t plane. In this procedure, we first establish the initial characteristic that depicts the motion of the wave front. The advancing characteristics are issued from the t-axis (upstream boundary) and the receding characteristics are issued from the initial characteristic. The network is extended to cover as much of the x,t plane as desired. The following are brief descriptions of the various phases of the procedure.

Numerical Procedure

Initial Characteristic

From point O ($x=0$, $t=t_o$) in Fig. 2, the initial characteristic C_o is drawn first. C_o is an advancing characteristic and the values of v and y (or c) are known upon it as functions of x because C_o represents times at which the disturbance originating at $x=0$ at $t=t_o$ reaches all points in the channel. The coordinates of points on C_o are determined by stepwise integration of equation (3). Thus,

$$\frac{x(M_2) - x(M_1)}{t(M_2) - t(M_1)} = v(M_1) + c(M_1). \qquad (7)$$

From this equation, $x(P_2)$ corresponding to a suitable value of $\Delta t = t(M_2) - t(M_1)$ can be computed.

Interior Characteristics

For the purpose of illustration, let it be assumed that computations have progressed to the line M_1J_1, with all characteristics to the left of this line being known. At each of the node points M_1, P_1, Q_1, $\cdots$, J_1, the slopes of the advancing characteristics can be calculated from the finite difference form of equation (3). By choosing an appropriate time interval Δt on the characteristic segment M_1M_2 so that $t(M_2) = t(M_1) + \Delta t$, and by solving for $x(M_2)$ from equation (7), the coordinates of M_2 on the x,t plane become determined. Thus M_2 becomes the starting point for establishing a receding characteristic. The intersection of an advancing characteristic drawn from P_1 and a receding characteristic from M_2 locates P_2 and the values of variables on P_2. The computation is then advanced to Q_2, R_2, $\cdots$, J_2 and finally to K_1, which lies on the t-axis and is a boundary point.

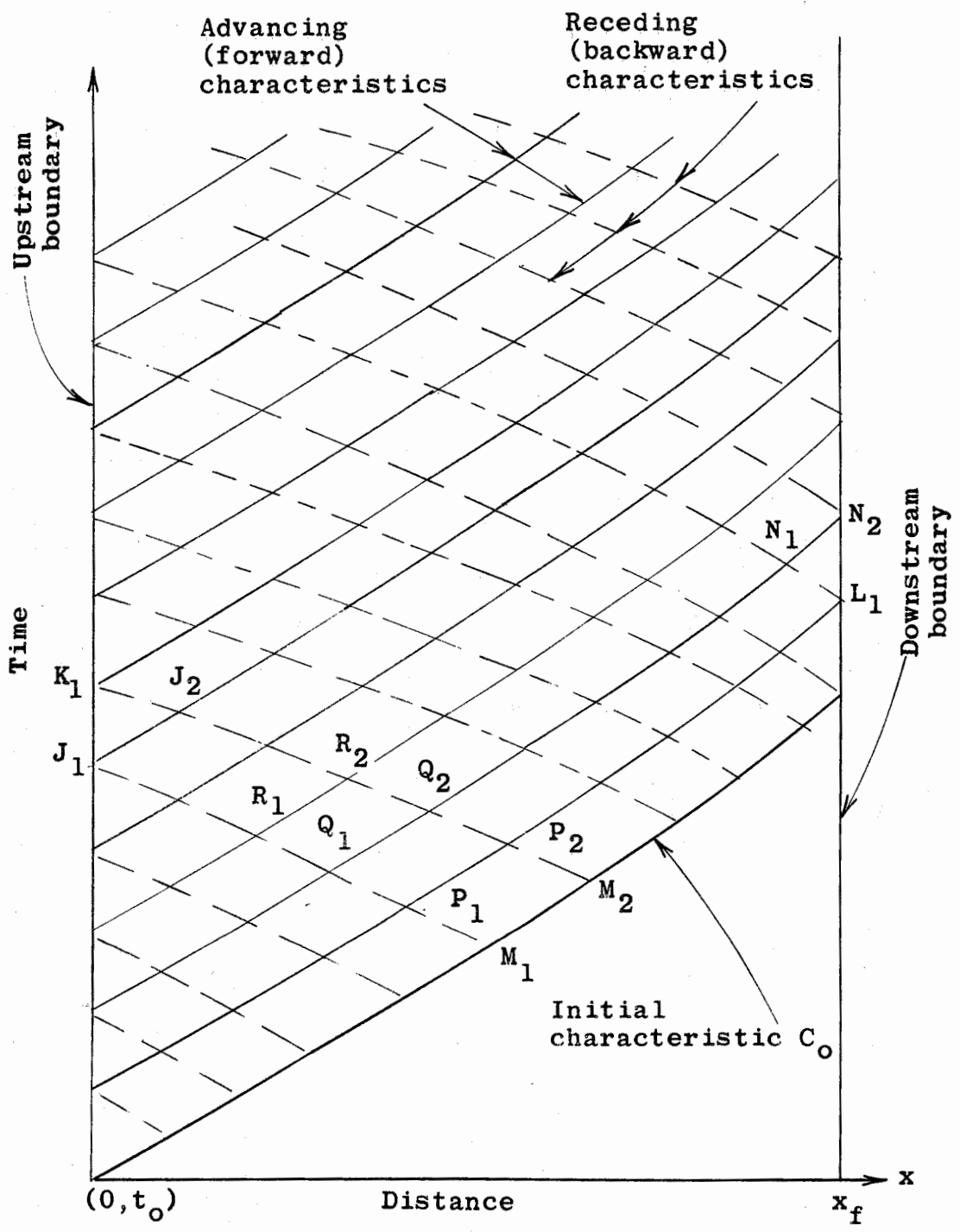


Fig. 2. Network of characteristics on the (x,t) plane

The computation of the variables at Q_2 is typical of all interior points and is outlined below. Since Q_2 lies on the advancing characteristic issued from Q_1, the equations of the advancing characteristic are applied to Q_1 and Q_2. Further, since Q_2 lies on the receding characteristic issued from P_2, the equations of the receding characteristic apply to P_2 and Q_2. Thus, we apply equations (3) and (5) in finite difference form to obtain the following system of equations.

$$x(Q_2) - x(Q_1) - [v(Q_1) + c(Q_1)][t(Q_2) - t(Q_1)] = 0 \qquad (8)$$

$$v(Q_2) - v(Q_1) + \sqrt{gB(Q_1,Q_2)/A(Q_1,Q_2)}\cdot[y(Q_2) - y(Q_1)]$$
$$- g[S_o - S_f(Q_1,Q_2)] - q(Q_1,Q_2)[x(Q_2) - x(Q_1)] = 0 \qquad (9)$$

$$x(Q_2) - x(P_2) - [v(P_2) - c(P_2)][t(Q_2) - t(P_2)] = 0 \qquad (10)$$

$$v(Q_2) - v(P_2) - \sqrt{gB(Q_2,P_2)/A(Q_2,P_2)}\cdot[y(Q_2) - y(P_2)]$$
$$- g\{[S_o - S_f(Q_2,P_2)]$$
$$- q(Q_2,P_2)[x(Q_2) - x(P_2)]\}\cdot[t(Q_2) - t(P_2)] = 0. \qquad (11)$$

In these equations,

$$A(Q_2,P_2) = \frac{1}{2}[A(Q_2) + A(P_2)]; \quad B(Q_2,P_2) = \frac{1}{2}[B(Q_2) + B(P_2)]; \cdots$$

These equations contain $v(Q_2)$, $c(Q_2)$, $A(Q_2)$, $B(Q_2)$, $S_f(Q_2)$, $t(Q_2)$ and $x(Q_2)$ as the unknowns. But since $A(Q_2)$, $B(Q_2)$ are functions of depth $y(Q_2)$ and $S_f(Q_2)$ depends on $v(Q_2)$ and $y(Q_2)$, there are only four independent unknowns, namely $x(Q_2)$, $t(Q_2)$, $v(Q_2)$, and $y(Q_2)$. We have four simultaneous nonlinear algebraic equations for evaluating the unknowns.

Upstream Boundary

Although the boundary conditions could be prescribed in several ways, depending on the type of problem to be solved, we assume for purposes of illustration that the upstream boundary is given by a stage hydrograph. Let us assume that we want to complete the computations on the node points of the characteristic ussed from M_2. Then, the values of the variables have been determined up to and including J_2 from the

calculations for interior points. From J_2 a receding characteristic can be drawn to meet the t-axis at K_1. At K_1, $y(K_1)$ is known from the stage hydrograph and $x(K_1) = x_o$. Two independent unknowns $t(K_1)$ and $v(K_1)$ are computed from the two equations of the receding characteristic applied to J_2 and K_1 We have, therefore,

$$x(K_1) - x(J_2) - [v(J_2) - c(J_2)]\cdot[t(K_1) - t(J_2)] = 0 \qquad (12)$$

$$v(K_1) - v(J_2) - \sqrt{g[B(K_1,J_2)/A(K_1,J_2)]}\cdot[y(K_1) - y(J_2)]$$

$$- g[S_o - S_f(K_1,J_2)]\cdot[t(K_1) - t(J_2)]$$

$$- q(K_1,J_2)[x(K_1) - x(J_2)] = 0. \qquad (13)$$

The values of $t(K_1)$ and $v(K_1)$ are determined from the two nonlinear algebraic equations (12) and (13).

Downstream Boundary

For the treatment of the downstream boundary condition, we consider a typical situation in subcritical flow where a control section is chosen as the downstream end of a reach. At this boundary, the relationship between the velocity and the depth is given by a rating curve or a calibration curve of some sort. We may assume that a relationship of the type

$$v(N_2) = F[y(N_2)] \qquad (14)$$

at point N_2 (see Fig. 2) is available. Point N_2 lies on an advancing characteristic issued from N_1, an interior point. We apply the equations of the advancing characteristic in finite difference form to the points N_1 and N_2 to obtain

$$x(N_2) - x(N_1) = [v(N_1) + c(N_1)]\cdot[t(N_2) - t(N_1)] \qquad (15)$$

$$v(N_2) - v(N_1) + \sqrt{g[B(N_1,N_2)/A(N_1,N_2)]}\cdot[y(N_2) - y(N_1)]$$

$$- g[S_o - S_f(N_1,N_2)]\cdot[t(N_2) - t(N_1)]$$

$$- q(N_1,N_2)\cdot[x(N_2) - x(N_1)] = 0. \qquad (16)$$

Equations (14) through (16) constitute a system of three simultaneous algebraic equations from which unknowns $v(N_2)$, $y(N_2)$, and $t(N_2)$ can be determined.

Solution Technique

We see that the numerical solution of the equations of unsteady flow by the method of characteristics leads to the solution of systems of nonlinear algebraic equations. The system consists of four equations at the interior points and may be reduced to two equations at the boundaries. In the scheme presented here, the equations for the slopes of characteristics at interior points are linear. Therefore, if we solve (8) and (10) for $x(Q_2)$ and $t(Q_2)$, we need to solve only two simultaneous nonlinear equations for the two unknowns $v(Q_2)$ and $y(Q_2)$. More accurate representations of the slopes of the characteristics would involve implicit equations, but we do not believe that any improvement in accuracy would justify the extra computations. We can, in general, reduce the computation to the solution of two simultaneous equations at all points. In the following, a discussion of the solutions at interior points is presented. The same approach can be used at the boundaries.

Nonlinear algebraic equations of the type encountered in the method of characteristics cannot be solved in a straightforward manner. We have found Newton's iteration to be the most suitable for the analysis of typical flood-routing problems. A brief description of the technique is presented as follows.

In equations (9) and (11) the unknowns consist of $A(Q_2)$, $B(Q_2)$, $v(Q_2)$, $y(Q_2)$, and $S_f(Q_2)$, only two of which are independent. We have, therefore, two equations with two unknowns. The iterative solution of these equations by Newton's method consists in assigning trial values to the unknowns and making corrections to the trial values in one or more iteration cycles. For the sake of illustration, let $A^{(k)}$, $B^{(k)}$, $y^{(k)}$, $v^{(k)}$, and $S_f^{(k)}$ be the values of the variables at point Q_2 in the k^{th} cycle of iteration. Our objective is to correct these values through the $(k+1)^{th}$ cycle. When the values at the k^{th} cycle are substituted in equations (9) and (11), the right sides of the equations will not vanish. The values at the right side are known as the residuals. Let the residuals be designated as $R_1^{(k)}$ and $R_2^{(k)}$. Their values are, from equations (9) and (11):

$$R_1^{(k)} = v^{(k)} + \sqrt{g\,[B^{(k)} + b]/[A^{(k)} + a]}\;[y^{(k)} - d] - \frac{gS_f^{(k)}}{2}\,h + i \qquad (17)$$

and

$$R_2^{(k)} = v^{(k)} - \sqrt{g\left[B^{(k)} + j\right] / \left[A^{(k)} + m\right]}\ \left[y^{(k)} - n'\right] - \frac{gS_f^{(k)}}{2}\ p + q, \tag{18}$$

where b, a, d, h, i, j, m, n', p, and q are constants whose values depend on the known values of the variables at P_1, Q_1, and Q_2. Also, $R_1^{(k)}$ and $R_2^{(k)}$ are different from zero because their values are not based on the true values of the variables but rather on values approximated through the k^{th} cycle of iteration. By Newton's method (Crandall, 1957),

$$R_1^{(k)} = \left(\frac{\partial R_1}{\partial v}\right)_k dv + \left(\frac{\partial R_1}{\partial y}\right)_k dy \tag{19}$$

and

$$R_2^{(k)} = \left(\frac{\partial R_2}{\partial v}\right)_k dv + \left(\frac{\partial R_2}{\partial y}\right)_k dy, \tag{20}$$

where

$$dv = v^{(k+1)} - v^{(k)}$$

and

$$dy = y^{(k+1)} - y^{(k)}. \tag{21}$$

The values of the variables corrected through the $(k+1)^{th}$ cycle of iteration are found from the simultaneous solution of (19) and (20). The corrections to the values of $y(Q_2)$ and $v(Q_2)$ become

$$v^{(k+1)} - v^{(k)} = \frac{R_1^{(k)} \frac{\partial R_2}{\partial y} - R_2^{(k)} \frac{\partial R_1}{\partial y}}{\frac{\partial R_1}{\partial y} \frac{\partial R_2}{\partial y} - \frac{\partial R_1}{\partial y} \frac{\partial R_2}{\partial v}} \tag{22}$$

and

$$y^{(k+1)} - y^{(k)} = \frac{R_2^{(k)} \frac{\partial R_1}{\partial v} - R_1^{(k)} \frac{\partial R_2}{\partial v}}{\frac{\partial R_1}{\partial v} \frac{\partial R_2}{\partial y} - \frac{\partial R_1}{\partial y} \frac{\partial R_2}{\partial v}}. \tag{23}$$

In equations (22) and (23), all the partial derivatives and residuals are evaluated at the k^{th} iteration step. These equations provide the values of the unknown approximated through the $(k+1)^{th}$ iteration. If it is desired to improve the given approximations, the procedure will be repeated until

$$|v^{(q+1)} - v^{q}| \leq \epsilon \quad \text{and} \quad |y^{(q+1)} - y^{(q)}| \leq \eta$$

are satisfied, where the expressions on the left are the differences between two successive approximations of the same variable, and ϵ and η are tolerance errors.

The numerical procedure proved to be rapidly convergent for applications to flood-flow problems. In several examples tested, seldom was it necessary to use more than three iteration cycles to complete the calculations within stringent tolerance limits.

The solution of the equations of unsteady flow by the method of characteristics has been programmed in FORTRAN 4 for operation on an IBM System/360. Applications of the method are given in a subsequent section of this report.

V. EXPLICIT METHOD

The finite difference equations of the fixed mesh direct explicit method are based on the primary differential equations of unsteady flow without going through some intermediate transformation as in the method of characteristics. The method is called explicit because the unknowns can be computed explicitly from the algebraic finite difference equations. Application of the explicit method to flood flows is mostly the outcome of the pioneering work of J. J. Stoker and his colleagues at the Courant Institute of New York University. The method uses a fixed mesh of points on the (x,t) plane so that the data are given and the solutions are obtained at fixed, predetermined locations and times. Detailed descriptions of the numerical solutions of the equations of unsteady flow are given by Isaacson et al. (1956). The present study is based directly on the approach given in that report and deviates little from it. However, we had to prepare completely new computer programs for our work. Only a brief outline of the numerical procedure will be presented here. The interested reader is referred to the cited reports for details.

Numerical Procedure

The numerical procedure for the solution of the equations of unsteady flow by the explicit method is essentially based on finding the values at a future time step by extrapolating from a previous time step subject to the laws of mechanics and the boundary conditions. For the sake of convenience, the description of the procedure will be subdivided into interior points and boundary points.

Interior Points

Let it be assumed that all the variables have been computed through time t. In Fig. 3, all quantities are known on row t. It is required to evaluate the variables at time $t+\Delta t$. The variables and their partial derivatives at a point M in row t can be expressed in terms of the values of the variables at the neighboring points by the following equations.

$$v(M) = \frac{1}{2}[v(R) + v(L)],$$

$$\frac{\partial A}{\partial t}(M) = \frac{A(P) - A(M)}{\Delta t},$$

$$\frac{\partial v}{\partial x}(M) = \frac{v(R) - v(L)}{2\Delta x},$$

$$\frac{\partial A}{\partial x}(M) = \frac{A(R) - A(L)}{2\Delta x},$$

$$\frac{\partial v}{\partial t}(M) = \frac{v(P) - v(M)}{\Delta t},$$

$$\frac{\partial y}{\partial x}(M) = \frac{y(R) - g(L)}{2\Delta t},$$

and

$$S_f(M) = \frac{S_f(R) + S_f(L)}{2}. \qquad (24)$$

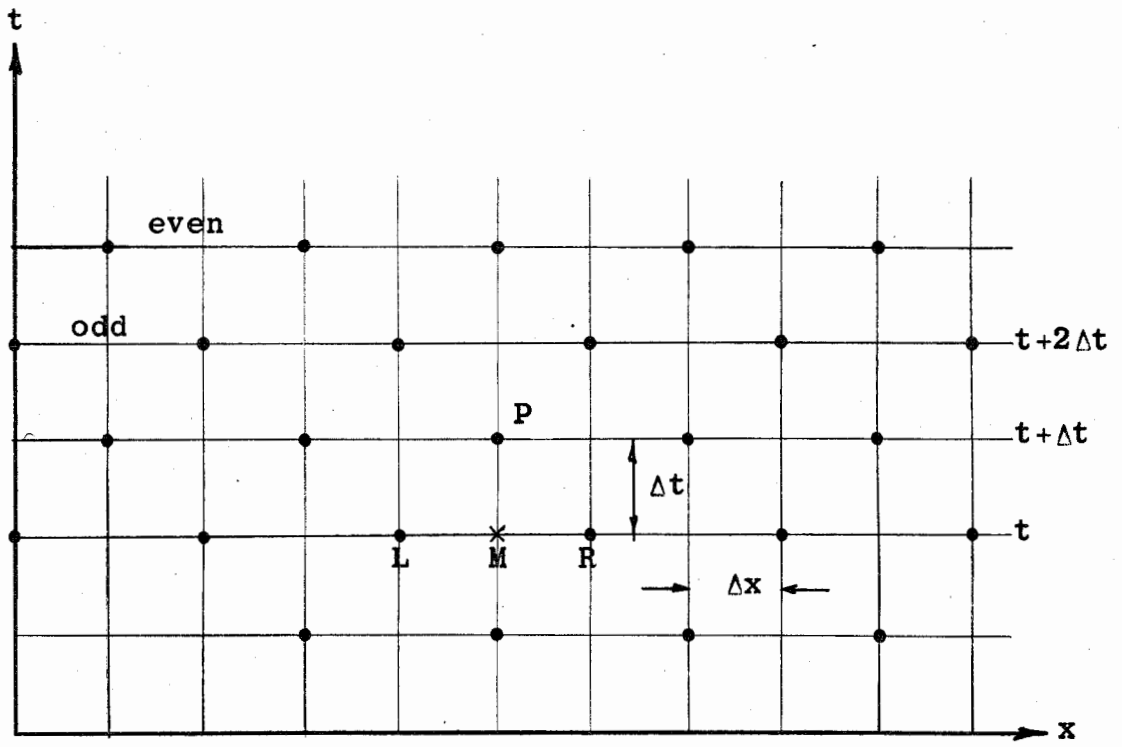


Fig. 3. Network of points on the (x,t) plane for the explicit method

The fundamental principles for the establishment of the numerical procedure for the explicit method are:

1. Applying the differential equations at a point where all the variables are known,

2. Using a centered difference scheme for the x-derivative, and

3. Using a forward difference scheme for the t-derivative.

It is seen that in this method the values of the unknowns at a time step $(t+\Delta t)$ are extrapolated from the known values at time t. When the functions and the partial derivatives in equations (1) and (2) are replaced by their finite difference analogues from equations (24), two equations for the determination of the unknowns v(P) and y(P) are obtained. They are:

$$v(P) = \frac{1}{2}[v(R) + v(L)] - \frac{\Delta t}{\Delta x}\left\{\frac{v(R)^2 - v(L)^2}{2}\right.$$
$$+ g[y(R) + q(R) - y(L) - q(L)]$$
$$\left. + g\left[\frac{S_f(R) + S_f(L)}{2}\right]\Delta x\right\}. \qquad (25)$$

$$y(P) = \frac{1}{2}[y(R) + y(L)] + \frac{2}{B(R) + B(L)}\frac{\Delta t}{\Delta x}[A(L)\cdot v(L)$$
$$- A(R)\cdot v(R)]. \qquad (26)$$

The relative grid size $\Delta t/\Delta x$ determines the stability of the computations. A necessary but insufficient condition for stability, generally known as the Courant stability condition, is

$$\Delta t \leq \frac{\Delta x}{|v + c|}. \qquad (27)$$

This relationship implies that the time step Δt must be so chosen that the point P lies within the area bounded by the characteristics issued from points L and R.

A staggered network of points is used in the scheme presented here. The rows may be labelled odd and even. It is noted that the quantities at point M do not enter the computations, so that M serves as a dummy point for establishing finite difference equations.

Upstream Boundary

At a point L (Fig. 4), one of the dependent variables, say the stage, may be known, or a relationship between the two dependent variables may be given, such as the value of the discharge. The complete determination of the boundary values requires a second relationship which can be obtained from the equations of unsteady flow. If it be assumed that the flow is subcritical and the stage (or depth) is known at a point L, the second relationship is given as (Isaacson et al., 1956)

$$\frac{\partial}{\partial x}(vA) + B\frac{\partial H}{\partial t} - q + \sqrt{BA/g}$$

$$\times\left(\frac{\partial v}{\partial t} + v\frac{\partial v}{\partial x} + g\frac{\partial H}{\partial x} + gS_f + \frac{qv}{A}\right) = 0, \qquad (28)$$

where H is the stage height.

Equation (28) is obtained by subtracting equation (2) from equation (1) after equation (2) is multiplied by $\sqrt{BA/g}$ to make both equations dimensionally compatible. Through this procedure equation (28) simulates the equation of a receding characteristic. Equation (28) is applied to point R (Fig. 4) in finite difference form. The following algebraic equation is found for computing the value of v(L).

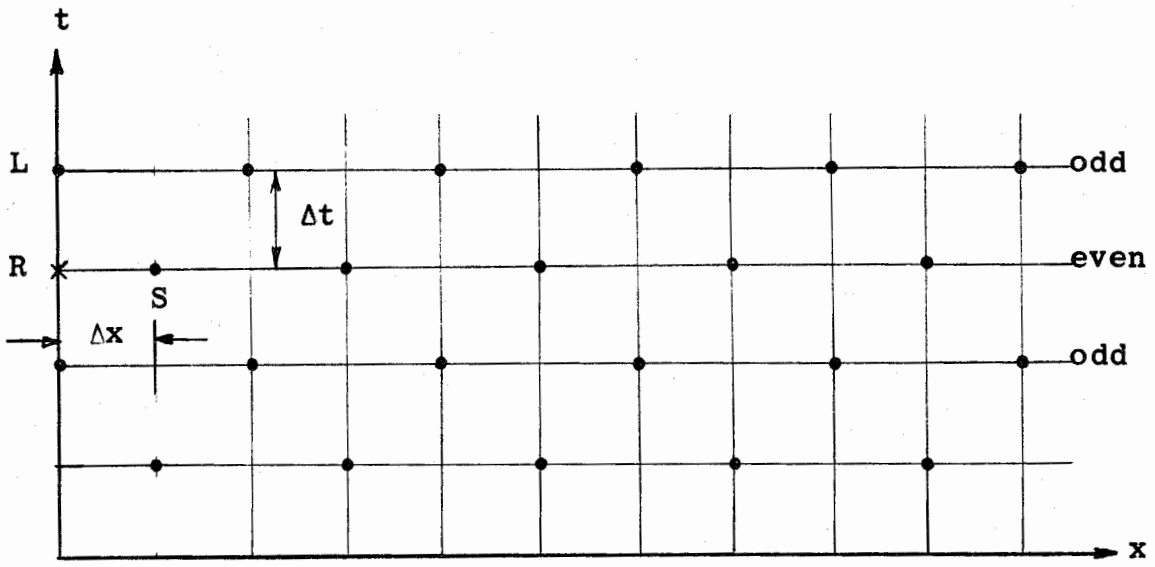


Fig. 4. Network of points on the (x,t) plane for computing on the upstream boundary by the explicit method

$$v(L) = v(R) - \Delta t\{v(L)^2 - v(R)^2 + \frac{g}{\Delta x}[H(S) - H(R)]$$

$$+ q\frac{v(R)}{A(R)} + q\cdot S_f(R)\} + \sqrt{g/B(R)A(R)}\cdot B(R)$$

$$\times [H(L) - H(R)]$$

$$+ \frac{\Delta t}{\Delta x}\sqrt{g/B(R)A(R)}\cdot[A(S)v(S) - A(R)v(R)]$$

$$- q\,\Delta t\sqrt{g/A(R)B(R)}. \qquad (29)$$

Downstream Boundary

For the solution of unsteady subcritical flow problems, it is desirable that a control section should be chosen as the downstream boundary. At a control section a well-defined relationship between the variables is supposed to exist. A rating curve, for instance, gives the discharge as a function of depth, say,

$$Q(N) = f[y(N)]. \tag{30}$$

Since there are two independent unknowns to be determined at the downstream boundary, a second relationship to supplement equation (30) is needed. Similar to the upstream boundary conditions, the second relationship can be given as (Isaacson et al., 1956)

$$\frac{\partial}{\partial x}(Av) + B\frac{\partial H}{\partial t} - q - \sqrt{BA/g}\left(\frac{\partial v}{\partial t} + v\frac{\partial v}{\partial x} + g\frac{\partial H}{\partial x} + gS_f + \frac{qv}{A}\right) = 0. \tag{31}$$

Equation (31) is obtained by subtracting equation (2) from equation (1) after equation (2) has been multiplied by $\sqrt{BA/g}$ to make the two equations compatible.

The finite difference representation of equation (31) is applied to the boundary point J (Fig. 5), which results in the following nonlinear algebraic equation:

$$\frac{B(J)}{\Delta t}[H(N) - H(J)] + \frac{A(J)v(J) - A(K)v(K)}{\Delta x} - q(J) + \sqrt{g[A(J)/B(J)]}\,\frac{v(N) - v(J)}{\Delta t} + \frac{v^2(J) - v^2(K)}{\Delta t} + \frac{g}{\Delta x}[H(J) - H(L)] + q(J)\frac{v(J)}{A(J)} + gS_f(J) = 0. \tag{32}$$

Equations (30) and (32) constitute a system of two simultaneous algebraic equations from which the unknowns v(N) and y(N) can be evaluated. Equation (32) simulates the equation of an advancing characteristic running from K to N.

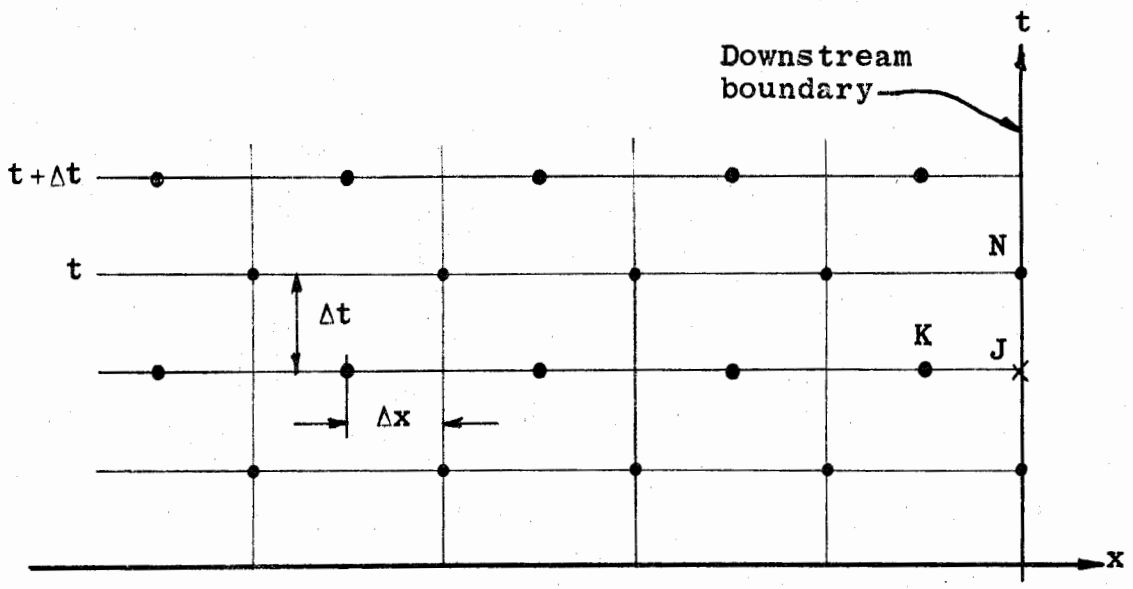


Fig. 5. Network of points on the (x,t) plane for computations at the downstream boundary

VI. IMPLICIT METHOD

The fundamental principle of the fixed mesh implicit method is the numerical solution of the equations of unsteady flow by establishing relationships among the variables simultaneously at all locations in the channel. Instead of determining the values of the variables at a future time step by extrapolating from known values from a previous time step, as is done in the explicit method, relationships existing among the variables at the future time step are used in the implicit method, the relationships being based on the equations of unsteady flow. In the scheme proposed in this report, a centered difference scheme is used to represent all partial derivatives, and the finite difference equations are found to constitute systems of nonlinear algebraic equations. This approach avoids the stringent stability condition of the explicit method and permits the use of large time steps for the solution of the equations. Although several investigations on the development of implicit methods have been reported in the literature (Liggett and Woolhiser, 1967; Abbott and Ionescu, 1967; Baltzer and Lai, 1968), none of them exhibited a real improvement over the corresponding explicit method. This report presents a procedure which is superior to the other methods, from the standpoint of both accuracy of results and computer time.

Numerical Procedure

The procedure for the numerical solution of the equations of unsteady flow is established over a discrete rectangular net of points on the (x,t) plane. The net points are determined by the intersection of straight lines parallel to the x and t-axes. The lines drawn parallel to the t-axis represent locations along the channel while those drawn parallel to the x-axis represent time. The location lines are drawn with spacing Δx, which need not be a constant. In this report, however, a fixed Δx is used for convenience. The t-axis may be used as the upstream channel boundary location, and the last line drawn parallel to the t-axis, to be designated the N^{th} line, can be used to represent the downstream channel boundary. The lines parallel to the x-axis are drawn with spacing Δt, which need not be constant. Each point of the network is identified by two indices, a subscript to designate the x-position of the point and a superscript to designate its t-value. Fig. 6 shows the network of points on the (x,t) plane to be used for the development of the numerical procedure.

As in other numerical methods, the numerical solution of the equations of unsteady flow by the implicit method is executed in two basic steps. The first step consists of

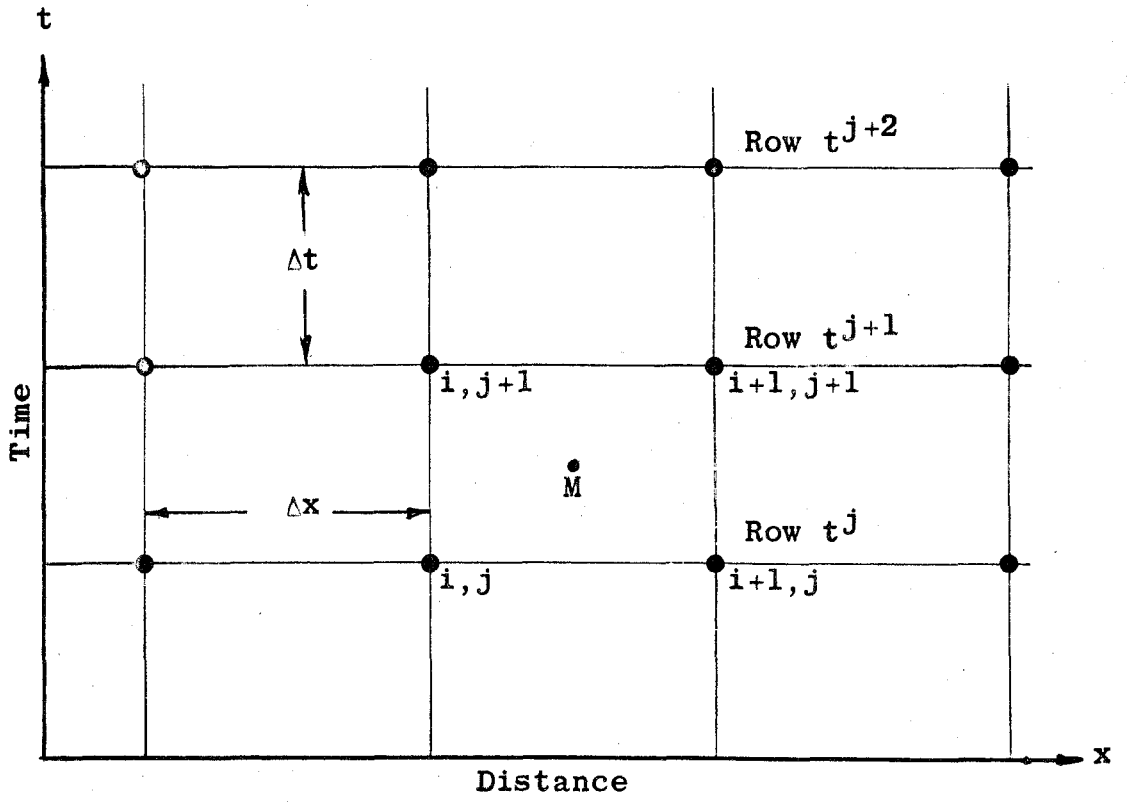


Fig. 6. Network of points on the (x,t) plane for the implicit method

representing the differential equations by a corresponding set of finite difference equations. The second step is the solution of the system of algebraic equations in conformance with the initial and boundary conditions.

The initial and boundary conditions necessary and sufficient for the solution of a problem by the implicit method are no different than for any other numerical method. As in the other methods described in this report, we assume that a stage or discharge hydrograph is known at a point $x = x_o$ at the upstream end of the channel. We may also assume that we know the values of the depth and velocity at all locations in the channel at an initial time $t = t_o$. The downstream boundary condition is assumed to be given by a control section so that a stage-discharge or a velocity-depth relationship is available.

The following procedure for the solution of the equations of unsteady flow by the implicit method is proposed. Let us assume that all the variables are known at all points of the network on the row t^j, i.e., time step t^j, and that it is

desired to find the values of the variables on row t^{j+1}; i.e., we advance the computations to time step $t^{j+1} = t^j + \Delta t$. Choosing a four-point grid identified by the intersections of the vertical lines x_i and x_{i+1} with the horizontal lines t^j and t^{j+1}, we apply the equations of unsteady flow in finite difference form to a point M centered within the four-point grid. This is in sharp contrast to the explicit method where the equations are applied to a point on the row t^j at which all the variables are known. At point M, we represent the average values and partial derivatives of a function α, as follows:

$$\alpha(M) = \frac{1}{4}(\alpha_i^j + \alpha_{i+1}^j + \alpha_i^{j+1} + \alpha_{i+1}^{j+1}), \tag{33}$$

$$\frac{\partial \alpha}{\partial x}(M) = \frac{1}{2\Delta x}[(\alpha_{i+1}^j + \alpha_{i+1}^{j+1}) - (\alpha_i^j + \alpha_i^{j+1})], \tag{34}$$

and

$$\frac{\partial \alpha}{\partial t}(M) = \frac{1}{2\Delta x}[(\alpha_i^{j+1} + \alpha_{i+1}^{j+1}) - (\alpha_i^j + \alpha_{i+1}^j)]. \tag{35}$$

In equations (33) through (35), α may represent v, y, A, S_f, B, so that v, y, A, S_f, $\partial v/\partial x$, $\partial y/\partial x$, $\partial v/\partial t$, $\partial y/\partial t$, B at point M are represented in terms of the values of the variables at the four corner grid points.

It should be noted that a centered finite difference scheme is used to represent both the x-derivative and the t-derivative. This approach also departs from other schemes where combinations of forward, centered, and backward differences are employed in the same manner. The term dA/dy is the width and may be represented by B. When y, v, $\partial v/\partial x$, $\partial v/\partial t$, ..., in the equations of unsteady flow (1) and (2) are replaced by their analogues from equations (33) through (35), the following two finite difference equations are obtained:

$$\frac{1}{2\Delta t}[(y_{i+1}^{j+1} + y_i^{j+1}) - (y_{i+1}^j + y_i^j)] + \frac{1}{2\Delta t} v_{i+1/2}^{j+1/2}[(y_{i+1}^{j+1} + y_{i+1}^j) - (y_i^{j+1} + y_i^j)] + \frac{1}{8\Delta x}(A/B)_{i+1/2}^{j+1/2}[(v_{i+1}^j + v_{i+1}^{j+1}) - (v_i^j + v_i^{j+1})] + (q/B)_{i+1/2}^{j+1/2} = 0 \tag{36}$$

and

$$\frac{g}{2\Delta x}[(y_{i+1}^{j} + y_{i+1}^{j+1}) - (y_{i}^{j} + y_{i}^{j+1})] + \frac{1}{2\Delta t}[v_{i+1}^{j+1} + v_{i}^{j+1}) - (v_{i}^{j} + v_{i+1}^{j})] + \frac{1}{2(\Delta x)} v_{i+1/2}^{j+1/2}[(v_{i+1}^{j+1} + v_{i+1}^{j}) - (v_{i}^{j} + v_{i}^{j+1})] + \frac{g}{4}(S_{f_i}^{j} + S_{f_{i+1}}^{j} + S_{f_i}^{j+1} + S_{f_{i+1}}^{j+1}) - \frac{g}{\Delta x}(z_{i}^{j} - z_{i+1}^{j}) + q(v/A)_{i+1/2}^{j+1/2} = 0, \quad (37)$$

where

$$v_{i+1/2}^{j+1/2} = \frac{1}{4}(v_{i}^{j} + v_{i+1}^{j} + v_{i}^{j+1} + v_{i+1}^{j+1}),$$

$$(v/A)_{i+1/2}^{j+1} = \frac{1}{4}[(v_{i}^{j}/A_{i}^{j}) + (v_{i+1}^{j}/A_{i+1}^{j}) + (v_{i}^{j+1}/A_{i}^{j+1}) + (v_{i+1}^{j+1}/A_{i+1}^{j+1})],$$

$$(q/B)_{i+1/2}^{j+1/2} = \frac{q}{4}[(1/B_{i}^{j}) + (1/B_{i+1}^{j}) + (1/B_{i}^{j+1}) + (1/B_{i+1}^{j+1})],$$

$$(A/B)_{i+1}^{j+1/2} = \frac{1}{4}[(A_{i}^{j}/B_{i}^{j}) + (A_{i+1}^{j}/B_{i+1}^{j}) + (A_{i}^{j+1}/B_{i}^{j+1}) + (A_{i+1}^{j+1}/B_{i+1}^{j+1})],$$

and the friction slope at a point (i,j), using Manning's formula, can be expressed as

$$S_{f_i}^{j} = \frac{(n_{i}^{j})^{2}(v_{i}^{j})\,|v_{i}^{j}|\,(B_{i}^{j})^{4/3}}{2.2082(A_{i}^{j})^{4/3}}.$$

In equations (36) and (37), all the variables with superscript (j) are known and all the variables with superscript (j+1) are the unknowns. However, all the unknowns are not independent. This is so because the area A and the top width B are functions of depth y, and the friction slope S_f is a function of depth y, velocity v, and friction factor n. These equations contain only four independent unknowns, the unknowns being the values of velocity and depth at points (i,j+1) and (i+1,j+1).

Equations (36) and (37) constitute a system of two nonlinear algebraic equations in four unknowns. By themselves, they are not sufficient to evaluate the unknown at points

(i,j+1) and (i+1,j+1). However, two unknowns are common to any two neighboring rectangular grids. Since there are N points on the row (j+1), there are (N-1) rectangular grids and (N-1) interior points. The combination of all rectangular grids provides 2(N-1) equations for the evaluation of 2N unknowns. Two additional equations that are needed for determining all the unknowns are supplied by the boundary conditions. If the stage at the upstream boundary is known as a function of time, then

$$y_1^{j+1} - f_1(t^{j+1}) = 0 \tag{38a}$$

becomes available as one supplementary equation. If the discharge is known there as a function of time, then

$$v_1^{j+1} \cdot A_1^{j+1} - Q_1^{j+1} = 0 \tag{38b}$$

becomes available as an alternate supplementary equation provided by the upstream boundary. If at the downstream boundary the discharge-stage relationship is known from the property of the control section or from a rating curve, then

$$y_N^{j+1} - f_N(v_N^{j+1}) = 0 \tag{39}$$

becomes available as the second supplementary equation provided by the downstream boundary. Therefore, the system of equations (26) through (39) provides 2N equations involving 2N unknowns, and the system becomes determinate.

In this system of equations, the values of the variables at time step t^j are known, and may be treated as constants. The unknowns consist of all the variables with superscripts (j+1). Equations (38a) and (38b) may be rewritten as

$$G_o(y_1) = y_1 - \lambda = 0 \tag{40a}$$

and

$$G_o(y_1, v_1) = v \cdot A_1 - Q_1 = 0, \tag{40b}$$

where λ is constant. Equation (39) may be rewritten as

$$F_N(y_N, v_N) = y_N - f_N(v_N) = 0 \tag{41}$$

Equations (36) and (37) may be rewritten as

$$F_i(y_i, v_i, y_{i+1}, v_{i+1})$$

$$= (y_{i+1} + y_i) + a + \frac{1}{4}\frac{\Delta t}{\Delta x}[(y_{i+1} - y_i)(v_i + v_{i+1} + b) + c(v_i + v_{i+1}) + d] + \frac{1}{4}\frac{\Delta t}{\Delta x}[(A_i/B_i) + (A_{i+1}/B_{i+1})](v_{i+1} - v_i + e) + \frac{1}{4}\frac{\Delta t}{\Delta x}[(h)\cdot v_{i+1} + m\cdot v_i + p] + \frac{\Delta t}{2} q[(1/B_i) + (1/B_{i+1}) + w] = 0 \quad (42)$$

and

$$G_i(y_i, v_i, y_{i+1}, v_{i+1})$$

$$= (y_{i+1} - y_i + a') - \frac{\Delta x}{g\Delta t}(v_i + v_{i+1} + b') + \frac{1}{4g}(v_{i+1}^2 + c'\cdot v_{i+1} + d'\cdot v_i - v_i^2 + e') + \frac{\Delta x}{2}(S_{f_i} + S_{f_{i+1}} + h') + \frac{\Delta x}{2g} q[(v_i/A_i) + (v_{i+1}/A_{i+1}) + m'] = 0, \quad (43)$$

where a, b, c, d, e, h, m, p, w, a', b', c', d', e', h', and m are all constants whose values depend on the values of the variables on row t^j. In equations (40) through (43), the superscripts have been dropped because all the subscripted variables now belong to row t^{j+1}.

We have, by means of equations (40) through (43) simulated the equations of unsteady flow, together with the boundary conditions, by a system of finite difference equations. The equations constitute a system of 2N nonlinear algebraic equations in 2N unknowns. For the sake of convenience, we assemble them below:

$$
\begin{aligned}
&G_p(y_1,v_1) = 0\\
&F_1(y_1,v_1,y_2,v_2) = 0\\
&G_1(y_1,v_1,y_2,v_2) = 0\\
&\quad\vdots\\
&F_i(y_i,v_i,y_{i+1},v_{i+1}) = 0\\
&G_i(y_i,v_i,y_{i+1},v_{i+1}) = 0\\
&\quad\vdots\\
&F_{N-1}(y_{N-1},v_{N-1},y_N,v_N) = 0\\
&G_{N-1}(y_{N-1},v_{N-1},y_N,v_{N-1}) = 0\\
&F_N(y_N,v_N) = 0.
\end{aligned}
\tag{44}
$$

Routine methods for the solution of nonlinear systems are not known. We propose to apply the generalized Newton's iteration method for the solution of this system. It should be noted that although the equations of system (44) involve 2N unknowns, each equation contains a maximum of four unknowns. This feature can be used to great advantage in the computational schemes.

The system of equations (44) is a special case of the general system of equations:

$$
\begin{aligned}
&\phi_1(\sigma_1,\sigma_2,\sigma_3,\cdots,\sigma_{2N}) = 0\\
&\phi_2(\sigma_1,\sigma_2,\sigma_3,\cdots,\sigma_{2N}) = 0\\
&\quad\vdots\\
&\phi_i(\sigma_1,\sigma_2,\sigma_3,\cdots,\sigma_{2N}) = 0\\
&\phi_{i+1}(\sigma_1,\sigma_2,\sigma_3,\cdots,\sigma_{2N}) = 0\\
&\quad\vdots\\
&\phi_{2N-1}(\sigma_1,\sigma_2,\sigma_3,\cdots,\sigma_{2N}) = 0\\
&\phi_{2N}(\sigma_1,\sigma_2,\sigma_3,\cdots,\sigma_{2N}) = 0,
\end{aligned}
\tag{45}
$$

where $\sigma_1, \sigma_2, \sigma_3, \cdots, \sigma_{2N}$ are the unknowns. For the sake of generality, a brief review of applying the generalized Newton iteration method to the system of equations (45) will be presented first. This will be followed by a more detailed application of the method to equations (44).

Generalized Newton Iteration Method

The application of the generalized Newton's iteration method (Ralston, 1965) to the system of equations (45) is made by assigning trial values to the unknowns. When these trial values are substituted into (45), the right sides of the equations may not vanish, but acquire values known as the residuals. Solutions are obtained by adjusting the trial values until each residual vanishes or is reduced to a tolerable quantity. The computation is organized in a series of iteration steps.

To illustrate this procedure, let us assume that the computations have been carried through the k^{th} iteration cycle, so that the values of the unknown have been approximated through the k^{th} cycle. Let the values of the variables at the end of the k^{th} cycle be denoted by the superscript k so that the values of the variables at the end of the k^{th} cycle are represented by $\sigma_1^k, \sigma_2^k, \cdots, \sigma_{2N}^k$. It is desired to approximate the values of the variables through the $(k+1)^{th}$ cycle. When the values of the variables approximated through the k^{th} cycle are substituted into equations (45), the right sides become the residuals. Let the residuals be represented by the vector R_i^k. Then,

$$\begin{aligned}
\phi_1(\sigma_1^k, \sigma_2^k, \cdots, \sigma_{2N}^k) &= R_1^k \\
\phi_2(\sigma_1^k, \sigma_2^k, \cdots, \sigma_{2N}^k) &= R_2^k \\
&\vdots \\
\phi_i(\sigma_1, \sigma_2, \cdots, \sigma_{2N}) &= R_i^k \\
&\vdots \\
\phi_{2N}(\sigma_1^k, \sigma_2^k, \cdots, \sigma_{2N}^k) &= R_{2N}^k.
\end{aligned} \tag{46}$$

According to the generalized Newton iteration method, the residuals and the values of the variables at two consecutive iteration cycles, (k) and (k+1), are related by the following euqations:

$$\frac{\partial \phi_1}{\partial \sigma_1} d\sigma_1 + \frac{\partial \phi_1}{\partial \sigma_2} d\sigma_2 + \cdots + \frac{\partial \phi_{2N}}{\partial \sigma_{2n}} d\sigma_{2N} = R_1^k$$

$$\frac{\partial \phi_2}{\partial \sigma_1} d\sigma_1 + \frac{\partial \phi_2}{\partial \sigma_2} d\sigma_2 + \cdots + \frac{\partial \phi_2}{\partial \sigma_{2N}} d\sigma_{2N} = R^k$$

$$\vdots$$

$$\frac{\partial \phi_{2N}}{\partial \sigma_{2N}} d\sigma_1 + \frac{\partial \phi_2}{\partial \sigma_2} d\sigma_2 + \cdots + \frac{\partial \phi_{2N}}{\partial \sigma_{2N}} d\sigma_{2N} = R_{2N}^k, \tag{47}$$

where all the partial derivatives are evaluated at the k^{th} cycle and

$$d\sigma_1 = \sigma_1^{k+1} - \sigma_1^k$$

$$d\sigma_2 = \sigma_2^{k+1} - \sigma_2^k$$

$$\vdots$$

$$d\sigma_{2N} = \sigma_{2N}^{k+1} - \sigma_{2N}^k. \tag{48}$$

Equations (47) constitute a system of 2N linear equations in 2N unknowns. The coefficients of the system are the values of the partial derivatives of the functions ϕ_i evaluated at the k^{th} cycle. The unknowns are $\sigma_1^{k+1}, \sigma_2^{k+1}, \ldots, \sigma_{2N}^{k+1}$. The solution of the linear system (47) provides the values of the variables approximated through the $(k+1)^{th}$ iteration cycle.

Since the system of equations (48) is linear, any of the well-known techniques for the solution of linear systems, such as the Gaussian elimination method or matrix inversion method, can be employed for its solution. Once the values of the variables at the $(k+1)^{th}$ iteration cycle have been determined, the procedure can be repeated to approximate the variables through succeeding iterations. The computations can be terminated whenever the difference between the values of any variable in two consecutive iteration cycles falls to below a tolerance limit. The values of the variables determined at the end of the terminal iteration cycle will be taken as the values for the time step (j+1), following which the computations will be advanced to the row (j+2), i.e., to time step (j+2).

The preceding discussion shows that in the generalized Newton iteration method the solution of a system of nonlinear equations is reduced to the successive solutions of systems of linear equations. The convergence of the procedure depends to a great extent on the choice of the initial trial values. The closer the trial values happen to be to the true values, the faster is the convergence. If the trial values are chosen in a completely arbitrary manner, the system may fail to converge. In streamflow problems, fortunately, excellent guides for the selection of trial values are available. We can choose the values from a preceding time step as the trial values, or better still, we can extrapolate from a preceding step.

Solution of the Finite Difference Equations

The finite difference equations based on the equations of unsteady flow as given by the system of equations (44) constitute a system of 2N nonlinear algebraic equations in 2N unknowns. These equations form a special case of the general system of nonlinear algebraic equations (45). The generalized Newton iteration method applicable to system (45) can be modified and streamlined for application to system (44).

As can be seen from equations (42) and (43), equations (44) fall into two main types, designated by the functions F_i and G_i. The functions F_i are derived from the equations of conservation of mass and the functions G_i are based on the conservation of momentum. Let the residuals of system (44) evaluated at the k^{th} iteration cycle be represented by $R^k_{1,i}$ and $R^k_{2,i}$, where R^k_1 is associated with function F and R^k_2 is associated with function G. The values of the residuals at the k^{th} iteration cycle are:

$$
\begin{aligned}
G_o(y_1^k, v_1^k) &= R^k_{1,0} \\
F_1(y_1^k, v_1^k, y_2^k, v_2^k) &= R^k_{1,1} \\
G_1(y_1^k, v_1^k, y_2^k, v_2^k) &= R^k_{2,1} \\
&\vdots \\
F_i(y_i^k, v_i^k, y_{i+1}^k, v_{i+1}^k) &= R^k_{1,i} \\
G_i(y_i^k, v_i^k, y_{i+1}^k, v_{i+1}^k) &= R^k_{2,i} \\
&\vdots \\
F_{N-1}(y_{N-1}^k, v_{N-1}^k, y_N^k, v_N^k) &= R^k_{1,N-1} \\
G_{N-1}(y_{N-1}^k, v_{N-1}^k, y_N^k, v_N^k) &= R^k_{2,N-1} \\
F_N(y_N^k, v_N^k) &= R^k_{1,N}
\end{aligned}
\qquad (49)
$$

The residuals and the partial derivatives of the system of equations (44) are related according to the generalized Newton iteration method by

$$\frac{\partial G_o}{\partial y_1} dy_1 + \frac{\partial G_o}{\partial v_1} dv_1 = R^k_{2,0}$$

$$\frac{\partial F_1}{\partial y_1} dy_1 + \frac{\partial F_1}{\partial v_1} dv_1 + \frac{\partial F_1}{\partial y_2} dy_2 + \frac{\partial F_1}{\partial v_2} dv_2 = R^k_{1,1}$$

$$\frac{\partial G_1}{\partial y_1} dy_1 + \frac{\partial G_1}{\partial v_1} dv_1 + \frac{\partial G_1}{\partial y_2} dy_2 + \frac{\partial G_1}{\partial v_2} dv_2 = R^k_{2,1}$$

$$\frac{\partial F_i}{\partial y_i} dy_i + \frac{\partial F_i}{\partial v_i} dv_i + \frac{\partial F_i}{\partial y_{i+1}} dy_{i+1} + \frac{\partial F_i}{\partial v_{i+1}} dv_{i+1} = R^k_{1,i}$$

$$\frac{\partial G_i}{\partial y_i} dy_i + \frac{\partial G_i}{\partial v_i} dv_i + \frac{\partial G_i}{\partial y_{i+1}} dy_{i+1} + \frac{\partial G_i}{\partial v_{i+1}} dv_{i+1} = R^k_{2,i}$$

$$\vdots$$

$$\frac{\partial F_{N-1}}{\partial y_{N-1}} dy_{N-1} + \frac{\partial F_{N-1}}{\partial v_{N-1}} dv_{N-1} + \frac{\partial F_N}{\partial y_N} dy_N + \frac{\partial F_N}{\partial v_N} dv_N = R^k_{1,N-1}$$

$$\frac{\partial G_{N-1}}{\partial y_{N-1}} dy_{N-1} + \frac{\partial G_{N-1}}{\partial v_{N-1}} dv_{N-1} + \frac{\partial G_N}{\partial y_N} dy_N + \frac{\partial G_N}{\partial v_N} dv_N = R^k_{2,N-1}$$

$$\frac{\partial F_N}{\partial y_N} dy_N + \frac{\partial F_N}{\partial v_N} dv_N = R^k_{1,N}, \qquad (50)$$

where

$$dy_1 = y^{k+1} - y^k_1$$
$$dv_1 = v^{k+1}_1 - v^k_1$$
$$\vdots$$
$$dy_i = y^{k+1}_i - y^k_i$$
$$dv_i = v^{k+1}_i - v^k_i$$
$$\vdots$$
$$dy_N = y^{k+1}_N - y^k_N$$
$$dv_N = v^{k+1}_N - v^k_N$$

and all the partial derivatives are evaluated at the k^{th} iteration cycle. The system of equations (50) is a system of 2N linear equations in 2N unknowns. Any of the standard methods, such as the Gaussian elimination method or the matrix inversion method, can be used for its solutions. The solution of the system will provide values of $y_1^{k+1}, v_1^{k+1}, \ldots, y_i^{k+1}, v_i^{k+1}, \ldots, v_N^{k+1}$, i.e., the values of the variables at the $(k+1)^{th}$ iteration cycle. The procedure will be repeated as many times as desired until the difference between the values of any variable in two consecutive iteration cycles falls below a tolerance value. The values of the variables found in the terminal iteration cycle will be taken as the values of the variables for the time step (j+1), and the computations will be advanced to time step (j+2).

A significant feature of the system of equations (50) is that the matrix of coefficients has a maximum of 4 nonzero elements in any row, because each equation involves at most only 4 of the 2N unknowns. The matrix is sparse and the nonzero elements are banded around the main diagonal. This property of the linear system can be used to great advantage in devising fast methods of solution.

In applications to problems, it becomes necessary to evaluate the coefficients of the linear system. The coefficients are the values of the partial derivatives of the functions F and G at each cycle of iteration. The expressions for the partial derivatives, which can be obtained by differentiating equations (42) and (43) with respect to the independent variables, are given below.

$$\frac{\partial F_i}{\partial y_i} = 1 - \frac{1}{4}\frac{\Delta t}{\Delta x}(v_i + v_{i+1} + b) + \frac{1}{4}\frac{\Delta t}{\Delta x}\left[1 - (A_i/B_i^2)(dB_i/dy_i)\right] - \frac{q}{2}\frac{\Delta t}{B_i^2}, \tag{51}$$

$$\frac{\partial F_i}{\partial y_{i+1}} = 1 + \frac{1}{4}\frac{\Delta t}{\Delta x}(v_i + v_{i+1} + b) + \frac{1}{4}\frac{\Delta t}{\Delta x}\left[1 - (A_{i+1}/B_{i+1}^2)(dB_{i+1}/dy_{i+1})\right] \cdot (v_{i+1} - v_i + e) - \frac{q}{2}\frac{\Delta t}{B_{i+1}^2}, \tag{52}$$

$$\frac{\partial F_i}{\partial v_i} = \frac{1}{4}\frac{\Delta t}{\Delta x}\left[(y_{i+1} - y_i) + c\right] - \frac{1}{4}\frac{\Delta t}{\Delta x}\left[(A_i/B_i) + (A_{i+1}/B_{i+1})\right] + \frac{m}{4}\frac{\Delta t}{\Delta x}, \tag{53}$$

$$\frac{\partial F_i}{\partial v_{i+1}} = \frac{1}{4}\frac{\Delta t}{\Delta x}\left[(y_{i+1} - y_i) + c\right] + \frac{1}{4}\frac{\Delta t}{\Delta x}\left[(A_i/B_i) + A_{i+1}/B_{i+1})\right] + \frac{1}{4}\frac{\Delta t}{\Delta x}\, h, \tag{54}$$

$$\frac{\partial G_i}{\partial y_i} = -1 + \frac{2}{3}(\Delta x)S_{f_i}\left[\frac{1}{B_i}\left(\frac{dB}{dy}\right)_i - \frac{B_i}{A_i}\right] - q\,\frac{\Delta x}{2g}\,v_i\,\frac{B_i}{A_i^2}, \tag{55}$$

$$\frac{\partial G_i}{\partial y_{i+1}} = 1 + \frac{2}{3}(\Delta x)S_{f_{i+1}}\left[\frac{1}{B_{i+1}}\left(\frac{dB}{dy}\right)_{i+1} - \frac{B_{i+1}}{A_{i+1}}\right] - q\,\frac{\Delta x}{2g}\,v_{i+1}\,\frac{B_{i+1}}{(A_{i+1})^2}, \tag{56}$$

$$\frac{\partial G_i}{\partial v_i} = \frac{1}{g}\frac{\Delta x}{\Delta t} - \frac{1}{2g}\,v_i + d' + \frac{\Delta x}{2v_i}\,S_{f_i} + \frac{q}{2g}\,\frac{\Delta x}{A_i}, \tag{57}$$

and

$$\frac{\partial G_i}{\partial v_{i+1}} = \frac{1}{g}\frac{\Delta x}{\Delta t} + \frac{1}{2g}(v_{i+1} + c') + \frac{\Delta x}{2v_{i+1}}\,S_{f_{i+1}} + \frac{q}{2g}\,\frac{\Delta x}{A_{i+1}}\,v_{i+1}. \tag{58}$$

VII. APPLICATIONS

Numerical flood routings of one form or another are extensively used in the planning, design, and management of water resources. The advent of the digital computer has made it feasible to obtain numerical integration of the equations of unsteady flow for the study of flood movements and other unsteady flows. At present, none of the methods based on the complete equations has become part of general practice. The explicit method has failed to achieve general acceptance, mostly because of its inefficiency. The accuracy of results obtained by it has also been subject to doubt. In spite of the fact that the use of numerical methods for unsteady flow investigations has increased tremendously in recent years, most of the investigations are still exploratory, and serious attempts at making them accessible to the average worker in the field have not yet been made.

A major task confronting the researcher is that of making the numerical methods accessible to the average user. These methods are based in the most part on rather sophisticated mathematical analysis for which the average user usually does not have adequate preparation to follow the details and to make modifications to the analysis for his own problem needs. Therefore, the researchers have not only to convince the user of the advantages or even of the need for using the complete equations, but also they should make the methods easily accessible to him. The preparation of computer programs requiring minimum or no modification except the insertion of input data would encourage the use of the newer methods. Most of the programs for unsteady flow that are currently available are research tools, and must be streamlined to fit the needs of different user categories. It is hoped that the programs prepared in conjunction with this study will be presented in a handy form. Prior to this, however, further testing of the programs with different types of problems and with a variety of boundary conditions is desirable.

In this report we concern ourselves with the application of the three methods presented in the preceding sections. There is great interest in applications to natural channels and in finding how accurately computed results agree with the observed values. We could select channels and provide this type of information. However, a study of that type is inconclusive for the following reason. Should there be a great discrepancy between the computed and the observed results, the discrepancy could not be all blamed on the mathematical model. The discrepancy could have been caused by errors in field measurement of the flows, by survey errors in the measurement of channel sections, by errors in estimating resistance coefficients, and most importantly by changes in the channel

properties before and during the unsteady flow event. To minimize these difficulties, the applications are conducted in two parts.

In the first part of the applications, the methods are applied to flow through an idealized channel. In doing this, we eliminate measurement errors, can concentrate on the numerical methods, compare the results of the different methods, and can reach useful decisions regarding their merits. In the second part, the methods are applied to a natural channel. The Neuse River, North Carolina, was chosen for this purpose because it is a major stream in North Carolina and many of its features are representative of the North Carolina rivers. Furthermore, its highly irregular features and its complex geometry provide a challenge to application of numerical methods. Unfortunately, complete and accurate data for the Neuse River, or for any other major North Carolina river, were not available. This part of the study, therefore, is essentially a study of the feasibility of applying the numerical methods to irregular natural channels, and is not a field verification of the mathematical models.

Computer Programs

Computer programs for the integration of the equations of unsteady flow for application to idealized and natural channels have been written in FORTRAN 4 for an IBM/System 360 Model 75. The programs are included in the Appendix to this report.

Unsteady Flow Through an Idealized Channel

The application of three numerical methods will be made to an idealized channel in order to compare the accuracy, the efficiency, and the convenience of the methods. The use of the idealized channel eliminates errors in flow measurement, in cross-sectional data, and in the estimates of channel resistance. The example selected for this purpose is a problem first posed by Harold Thomas (1934) for testing the validity of numerical methods.

The example invented by Thomas is the propagation of a flood in a uniform channel of infinite width with a bottom slope of 1 ft/mi. The channel is assumed to be 500 mi long, the upstream boundary condition is given by a discharge hydrograph of sinusoidal shape having a duration of 4 days and a peak flow per unit channel width of 200 ft^3/sec. The downstream condition is given by a stage-discharge relationship based on uniform flow. The initial conditions are given by a uniform flow per unit channel width of 50 ft^3/sec at a depth of 13.086 ft. The value of "n" in Manning's formula turns out

to be slightly <0.03. In the following paragraphs, solutions of this problem by the following methods will be discussed: (1) approximate, (2) explicit, (3) characteristic, and (4) implicit methods.

Approximate Method

Thomas used two approximate methods, one of which was the storage-routing method, for the solution of the problem. The grid sizes used were combinations of Δx = 25 mi, 50 mi and Δt = 12 hr, 24 hr, and 48 hr. It was found that peak flood flows and their times of arrival at downstream stations depended on the values of Δx and Δt. Thus, using Δx = 25 mi and Δt = 48 hr, the flood peak arrived at the 200 mi station in 48 hr and was reduced by 25%, while using Δx = 25 mi and Δt = 12 hr, the flood peak arrived at the same location in 29 hr and was reduced by only 4%. Similar results were obtained recently by Brakensiek and Comer (1966) when the example was worked out on an IBM 1620 computer using the same grid dimensions.

It is obvious that numerical methods utilizing large grid sizes are not reliable. Perhaps the basic concept of representation by finite differences is violated. It should be pointed out that when such methods are used in conjunction with field observations, false conclusions as to the "verification" of the methods may be reached. What may appear as field verification may actually be the selection of those values of Δx and Δt supplying a fictitious hydrograph that happens to agree with the recorded values.

Explicit Method

In the application of the explicit method to the given problem, the numerical scheme proposed by Stoker et al. (1953) was closely adhered to, except that the boundary conditions are different in this problem than in the problems solved by them. For the given problem, computations were made using combinations of Δx = 5 mi, 10 mi and Δt = 0.05 hr, 0.10 hr, and 0.15 hr. Stage hydrographs at 100 mi and 300 mi stations are plotted in Fig. 7. The stability condition requires that $\Delta t \leq 0.10$ hr for Δx = 5 mi and $\Delta t \leq 0.20$ hr for Δx = 10 mi. Thus, computations become unstable when Δt = 0.15 hr and Δx = 5 mi are used. It is seen that as the grid size increases, the accuracy drops off. However, almost identical times of peak flow are given by all $\Delta x, \Delta t$ relationships. It is clear that the accuracy of the computations is improved by using small values of Δx and Δt; however, this results in long computation times. When this problem was computed on an IBM 1410 system, many hours were needed for completing the work. This method did not appear to be feasible for routine

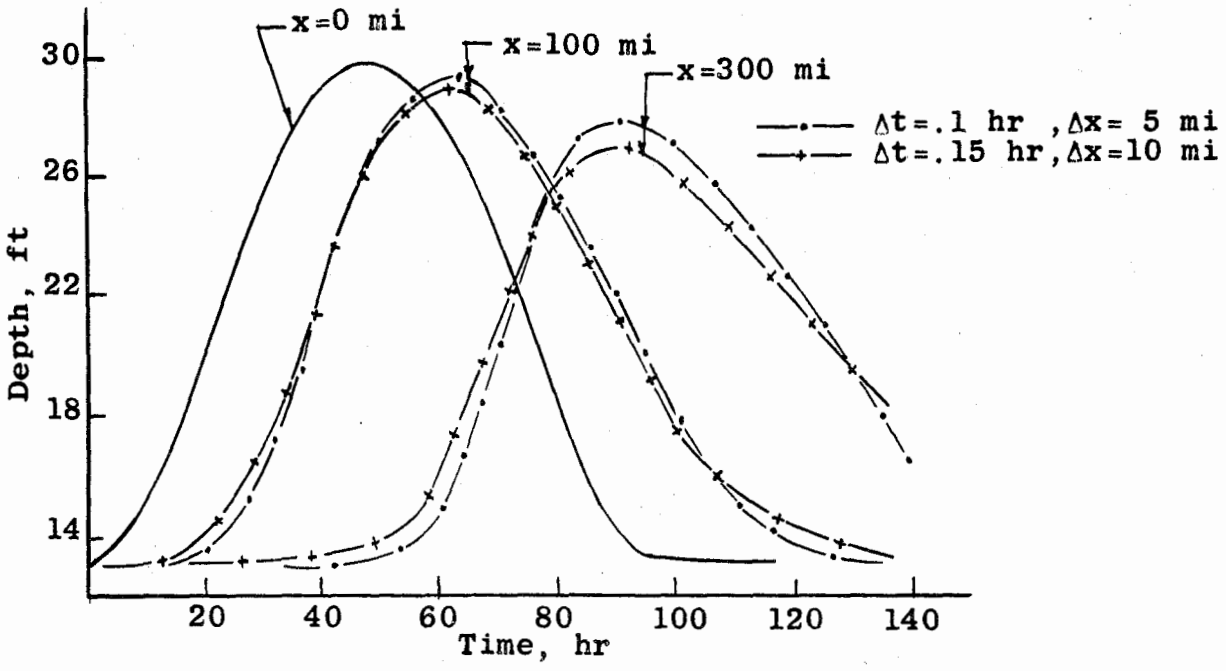


Fig. 7. Stage hydrographs by the explicit method

applications, since the use of time steps of a few minutes for flows with durations of days is not realistic. The situation has changed considerably in favor of the method with the introduction of the IBM/360 system.

Characteristic Method

In the application of the characteristic method to the given problem, values of 0.2 hr, 0.8 hr, 1.6 hr, 2.43 hr, and 3.24 hr were used for Δt. This method is stable for fairly large grid size, although the accuracy drops off as the grid size increases. This is clearly shown in Fig. 8, in which stage hydrographs at 100 mi and 300 mi stations are displayed. The main disadvantage of this method is that considerable interpolation becomes necessary when presenting the results. It becomes very awkward for natural channels where the geometry changes from section to section. Using Δt values of < 0.8 hr will not significantly improve the accuracy but will result in excessive computation times.

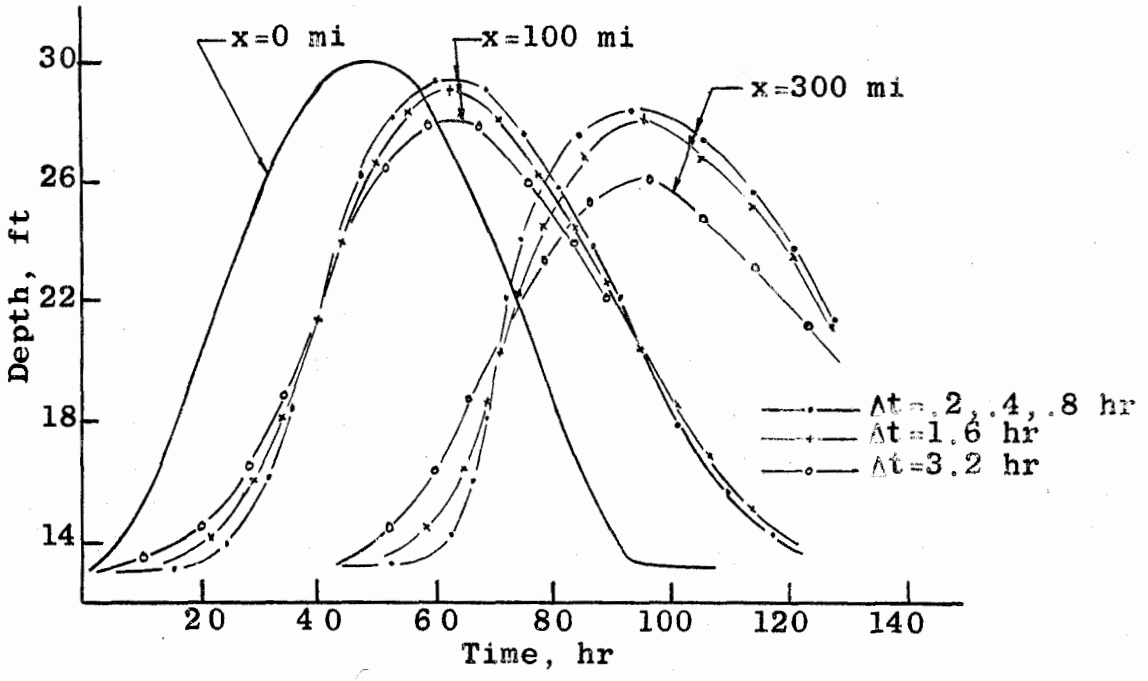


Fig. 8. Stage hydrographs by the method of characteristics

Implicit Method

In the application of the implicit fixed mesh method to the given problem, values of 5 mi and 10 mi for Δx and 0.5 hr, 1.0 hr, and 2.0 hr were used for Δt. Using $\Delta x = 5$ mi, it becomes necessary to solve 202 nonlinear algebraic equations simultaneously. When Δx is doubled, the number of equations is reduced by 50%. For $\Delta t = 0.5$ hr, computations can be completed with a single cycle of iteration. With $\Delta t = 1.0$ hr, several cycles of iteration become necessary to complete the computations within the tolerance limit. This limit was set to keep the difference in the velocity between two iteration cycles to <0.02 ft/sec. The results of the computations are displayed in Fig. 9 as stage hydrographs at 100 mi and 300 mi stations. It is noted that almost identical (to the second decimal place) results are obtained for $\Delta t = 0.5$ hr and 1.0 hr, with Δx kept at 5.0 mi.

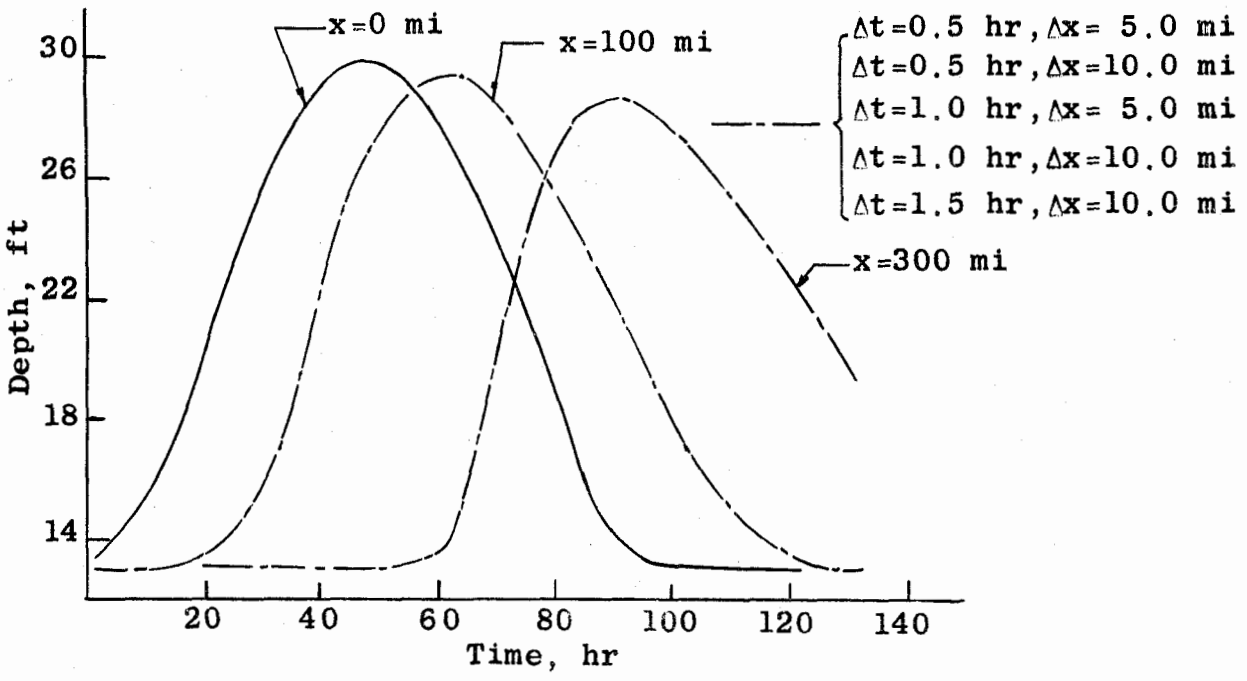


Fig. 9. Stage hydrographs by the implicit method

Comparison of Results and Conclusions

Although the numerical methods for flood routing discussed in this paper are all based on the same basic differential equations of unsteady flow, yet the algebraic representation used in each method is different. The question arises, therefore, that since we are dealing with different algebraic systems, different solutions of the primary equations may be obtained. The algebraic simulations are not useful unless they converge to the same solution. The results of the computations by the various methods can be used to provide the answer to this question.

The stage hydrographs by the three complete methods at 100 mi and 300 mi stations, using the smallest grid size in each case, are compared in Fig. 10. It is seen that times of arrival of flood peaks as given by all three methods are almost the same, and the depth values are not appreciably different.

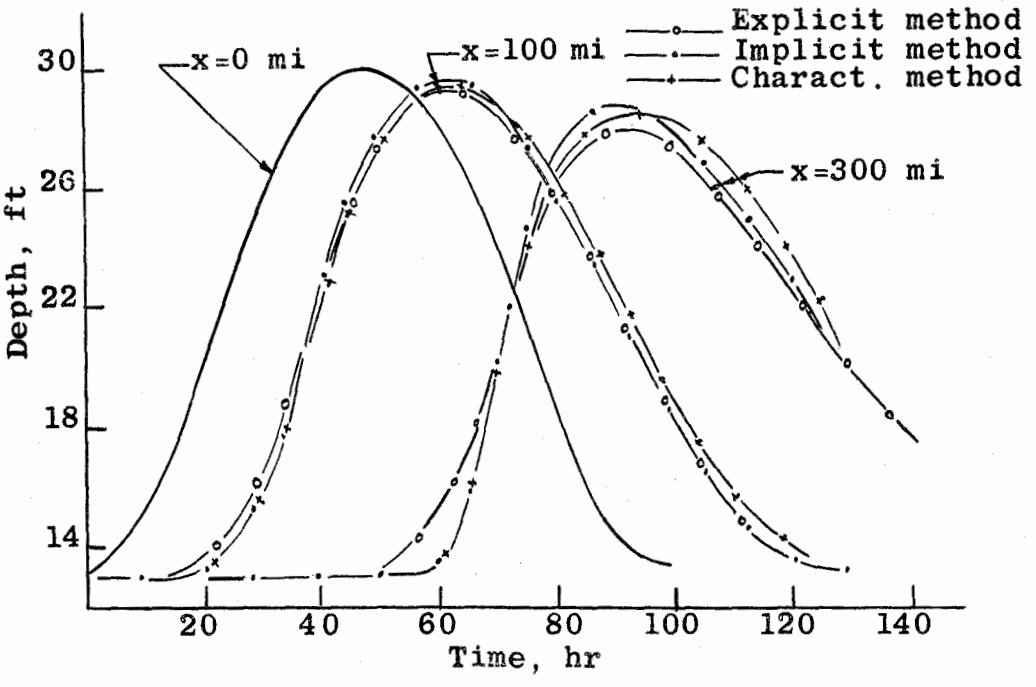


Fig. 10. Comparison of hydrographs by three complete methods

The implicit and the characteristic methods are probably more accurate than the explicit method, and the results obtained by the explicit method will be closer to the other two when smaller grid sizes are used. It also is clear that to obtain reliable results, small values of Δx and Δt, in the order of a few miles and preferably <1.0 hr, must be used. Such values, of course, will change from one problem to another. The use of large values of Δx and Δt, say 50 mi and 24 hr for the given example, will lead to erroneous results by any method. The high-speed digital computer with a large memory is an indispensible tool for numerical flood routing.

Of the three complete methods, the explicit method is the least efficient but is the simplest. The characteristic and implicit methods are both efficient; however, the characteristic method is the least convenient. The implicit method is both efficient and convenient. The study has shown that numerical methods require that small grid sizes, the size governed by the magnitudes of the variables involved, must be used to obtain accurate and reliable solutions. If a high-speed computer of large memory must be used, there is little to be gained by using the approximate methods. These latter methods will, of course, provide good results in some cases, while they may fail in others. Once the limitations and difficulties of the numerical

methods are known, it becomes meaningful to seek for verification of the methods in the field.

Unsteady Flow Through a Natural Channel

To test the feasibility of applying the numerical integration of the complete equations of unsteady flow to real problems and to investigate the flexibility, efficiency, and accuracy of different methods, flow through a natural, irregular river channel was analyzed. A channel reach was chosen for which adequate field hydrographic data were available. These data included a flood stage hydrograph, topographic maps, and other necessary gauging and channel information. The Neuse River, between Goldsboro, North Carolina, and Kinston, North Carolina, was found to be very suitable for this purpose. The flood data of September, 1945, and October, 1964, collected by the U.S. Geological Survey, were used in solving the unsteady flow equations. Measurements had been made during a 15-day period in 1945 and a 20-day period in 1964 along a 45 mi reach of the Neuse from Goldsboro to Kinston. Fig. 11 is a sketch of that part of the Neuse River under study, together with the reaches and gauging stations. The hydrographs at Goldsboro are shown in Fig. 12.

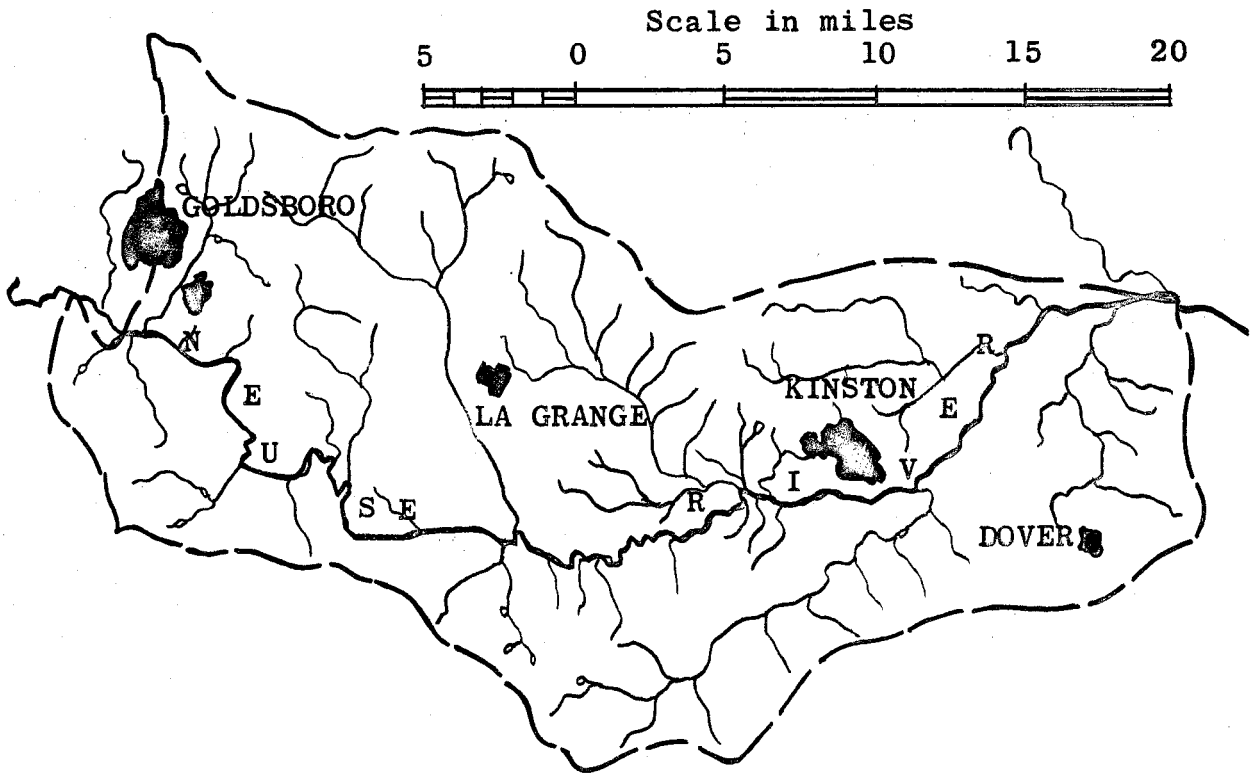


Fig. 11. Neuse River between Goldsboro and Kinston, North Carolina

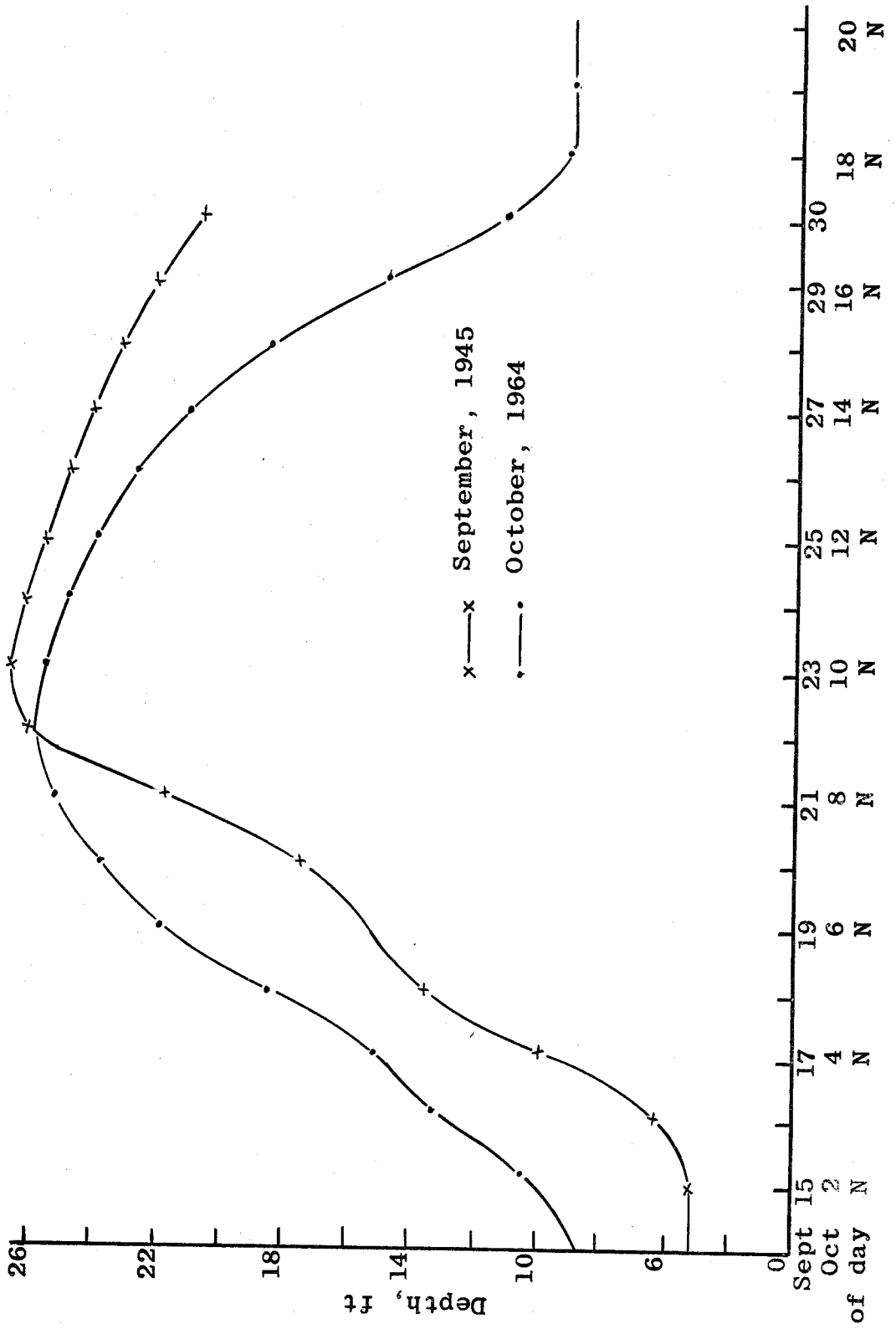


Fig. 12. Measured flood hydrographs at Goldsboro, North Carolina

In application to natural channels, the channel geometry is simulated by two different models, both of which are based on actual topographic data. In the first model, to be referred to as the composite channel section, the channel sections over the entire reach are represented by a single section derived by averaging channel properties over the reach. An average bottom slope based on the elevations of the channel bottom at the two ends of the reach is used in the computations.

In the second model, to be referred to as the discrete channel section, the actual values of the channel geometry are used in the computations. Thus the channel slope varies from section to section over the reach length.

The need for the composite section model arises because the explicit and the characteristic methods fail to work unless the changes in the channel geometry from section to section are gradual. Thus it is not feasible to introduce a waterway in which the bottom elevation changes significantly from section to section. Of the three methods studied, only the implicit method can handle channel geometries with significant changes in area, width, and bottom elevation from one section to the other.

Initial and Boundary Data

Initial depths were supplied by backwater calculation starting from a downstream depth. A linear interpolation was applied to the two initial discharges at the boundary points to calculate the initial discharges at all intermediate net points. The discharge values at the intermediate net points divided by the local cross sections gave the initial velocities. At the downstream boundary, a rating curve was used (Fig. 13). At the upstream end, the stage height was prescribed as a function of time.

For the 1945 flood, the initial values are given by an unsteady nonuniform flow with a depth of 5.17 ft at Goldsboro and 6.85 ft at Kinston at time $t = 0$ (flood data starting at 3:00 am, September 18, 1945). For the 1964 flood, the initial values are given by an unsteady nonuniform flow with a depth of 8.9 ft at Goldsboro and 5.9 ft at Kinston at time $t = 0$ (flood data starting at 3:00 am, October 2, 1964). In the 1945 flood, the depth of the water at point $x = 0$ (upstream boundary) as given by the hydrograph increased from 5.17 ft to 26.72 ft and then decreased to 19.8 ft within 360 hr. In the 1964 flood, the depth of the water increased from 8.9 ft to 26.0 ft and then decreased to 9.25 ft within 450 hr.

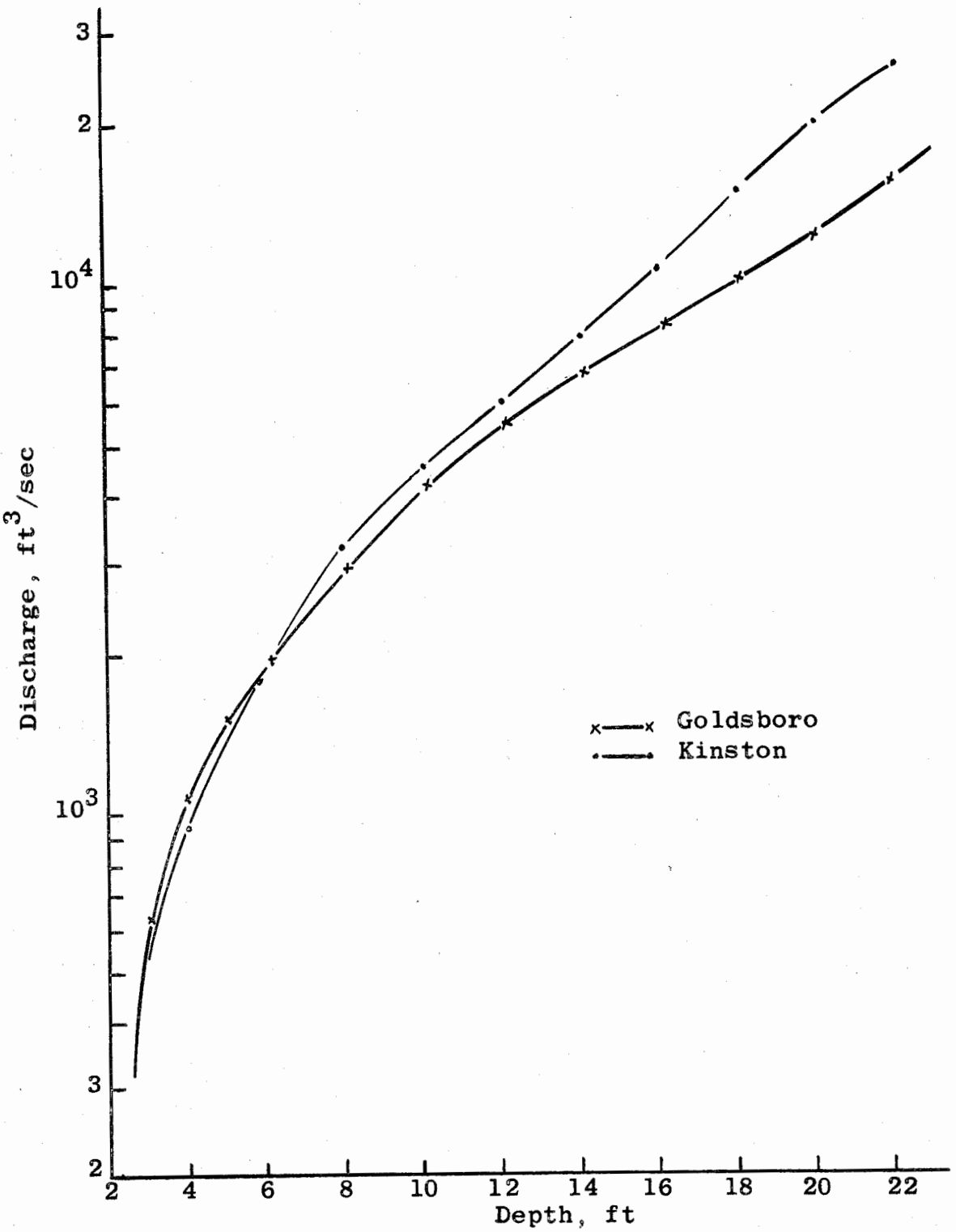


Fig. 13. Rating curves at Goldsboro and Kinston, North Carolina, in 1945

Computations for the Composite Section Channel

Determination of Channel Section Parameters. In the composite natural channel model, only one typical channel cross section A and one typical top width B were used to represent the whole reach. The typical cross-sectional areas as well as top widths were taken from the average of sixteen individual channel sections (Fig. 14) which were calculated from the U.S. Geological Survey topographic maps (1902, 1942, 1957a,b, 1958b). The width B was calculated from the relation, $B = dA/dy$.

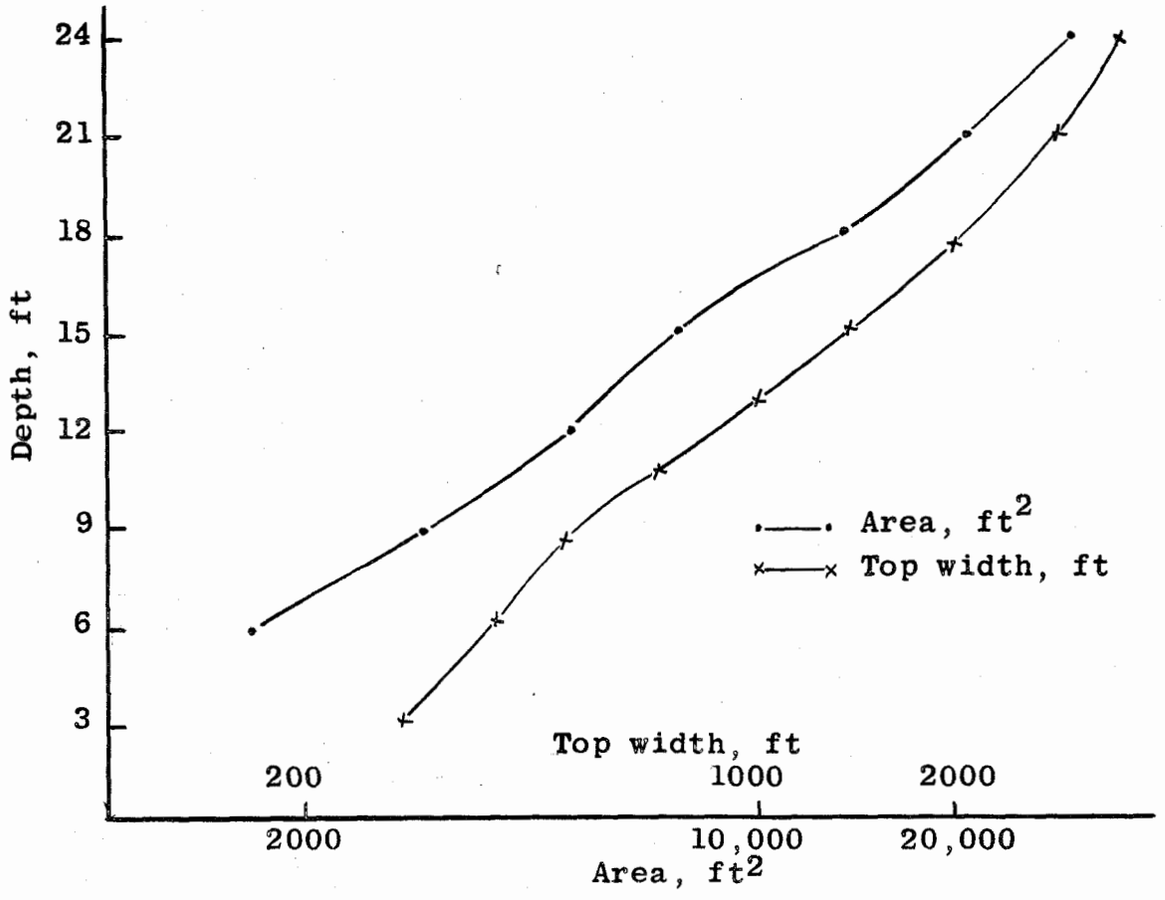


Fig. 14. Composite section area and top width

Channel Bottom Slope. The channel bottom slope S was assumed constant and was calculated from the formula,

$$S = \frac{\text{Gauging station datum height at Goldsboro} - \text{Gauging station datum height at Kinston}}{\text{Total river length from Goldsboro to Kinston}}$$

Manning's Roughness Coefficients. Two polynomial equations in terms of depth y representing Manning's roughness coefficients were determined at both the upstream and downstream ends. At intermediate net points a linear interpolation with respect to the changes in x-distance was used to determine all the Manning's roughness coefficients. Manning's roughness coefficients n were calculated by using Manning's equation,

$$n = 1.486R^{2/3}S^{1/2}/v. \qquad (59)$$

In evaluating n, two factors were taken into consideration: (1) the variation of n with depth at a given section and (2) the variation of n from section to section. Polynomial equations giving the variation of n with depth at the downstream and upstream boundaries were determined from rating curves and past records. At intermediate points the value of n was assumed to vary linearly between the two ends for the same depth. Manning's roughness coefficient n varies from 0.04 to 0.11, increasing with respect to depth as the depth changes from 5.17 ft to 27 ft at the upstream end. At the downstream end, it varied from 0.039 to 0.072, corresponding to depths of from 6 ft to 24 ft, respectively. In Fig. 15, Manning's roughness coefficient n is plotted versus the depth at Goldsboro and Kinston, respectively.

Lateral Inflow q. Lateral inflow q is a function of location x and time t, but it is difficult to measure. In the present study, lateral inflow q was assumed to be a constant along the tributaries. The value $q = 0.005$ to $q = 0.02$ $ft^3/sec/ft$ was suggested by U.S. Geological Survey hydrologists on the basis of their knowledge of the Neuse River.

Results of Computations. Computations for predicting the 1945 flood hydrograph at Kinston from the input data at Goldsboro were made by three methods. In the explicit method, a net with increments $\Delta x = 1.5$ and 3.0 mi in the x-direction along the channel and $\Delta t = 0.025$ and 0.03 hr in the t-direction were used. Larger Δt's can be used initially, but when the water depth increases to 20 ft, smaller values of Δt (≤ 0.025 hr) should be used or the computation becomes unstable.

In the method of characteristics, constant Δt's were assumed and the Δx's were determined by the corresponding Δt's. Constant time increments $\Delta t = 0.1$, 0.5, 0.75, and 1.0 hr gave almost identical results. When larger Δt's were tested, some irregular oscillations showed up.

In the implicit method, $\Delta x = 3$ mi and time increments $\Delta t = 0.15$, 1, 2, 5, 10, and 20 hr were tested.

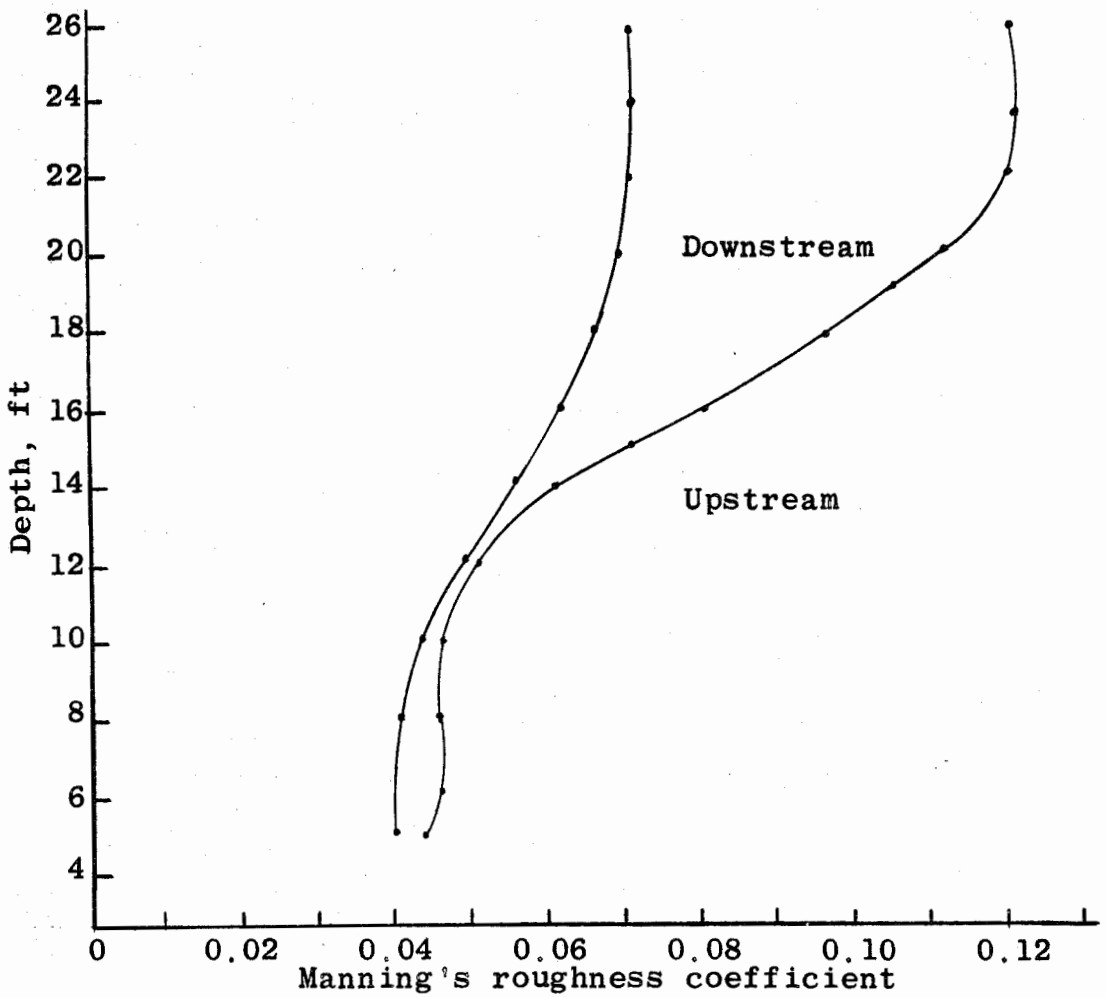


Fig. 15. Manning's roughness coefficient at Goldsboro and Kinston, North Carolina, for the flood in 1945

Fig. 16 shows the depths computed by the explicit, implicit, and characteristic methods for the 1945 flood. Time increment Δt in the method of characteristics is 0.25 hr; in the explicit method, 0.025 hr; and in the implicit method, 5 hr.

Fig. 17 shows the results for different methods of calculation for the 1964 flood. The calculated results were obtained by using the same coefficients as were used to calculate the 1945 flood.

The effects of different time increments, lateral inflows, roughness coefficients, and initial and boundary conditions on the outflow hydrographs were investigated for the 1964 flood, and they showed similar effects as for the 1945 flood.

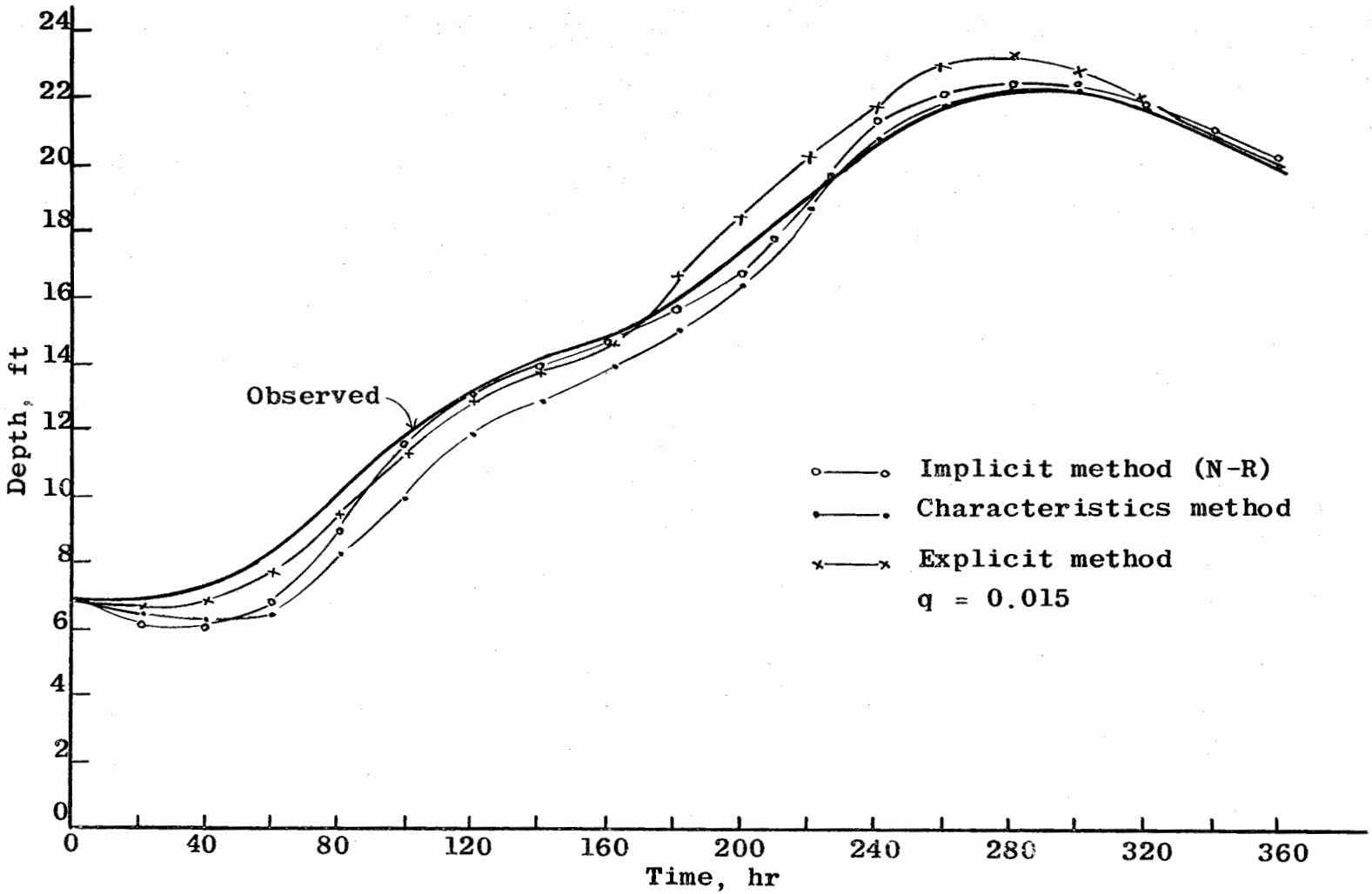


Fig. 16. Calculated 1945 flood hydrograph near Kinston, North Carolina (composite section)

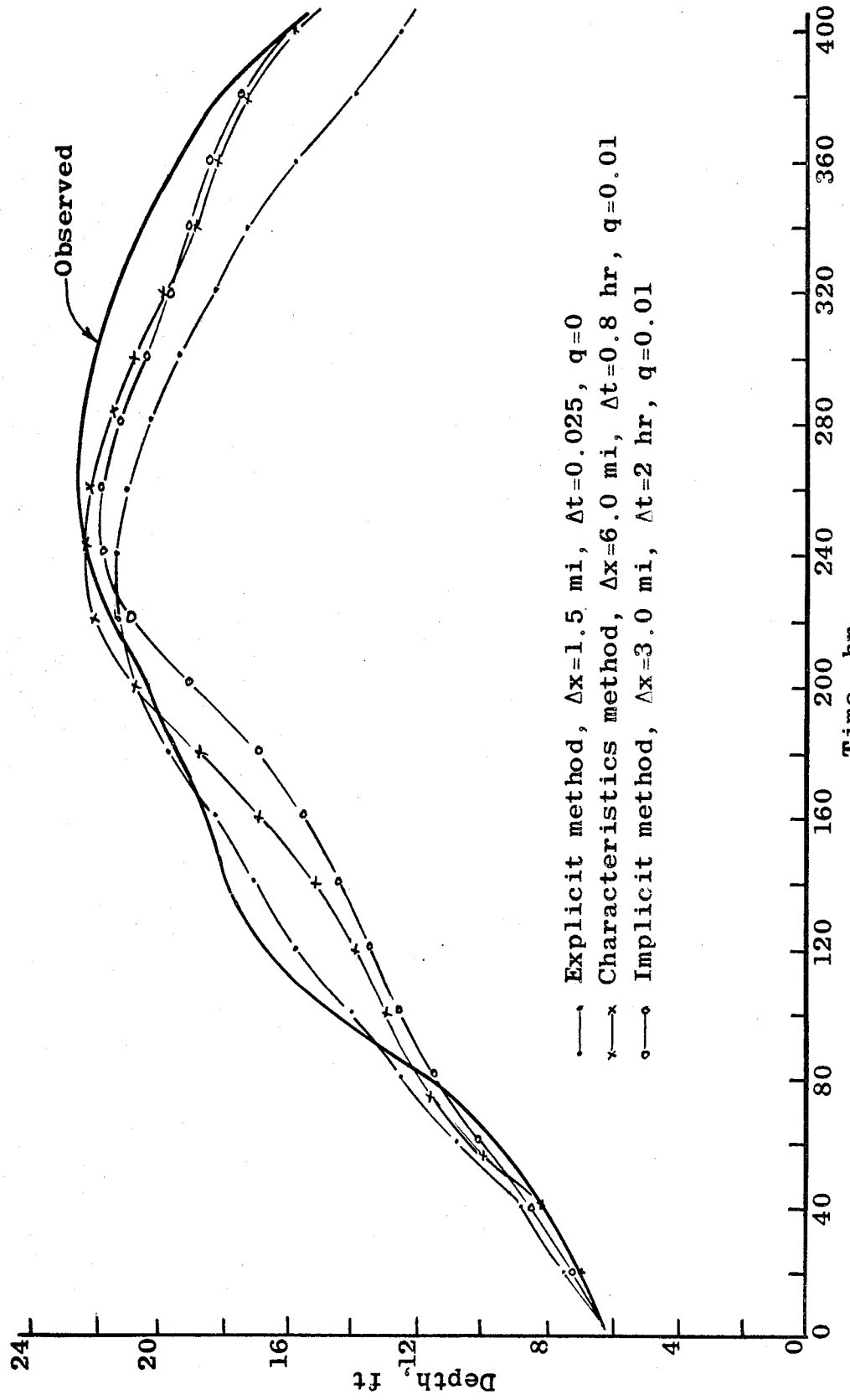


Fig. 17. Calculated 1964 flood hydrograph near Kinston, North Carolina (composite section)

Computations for Discrete Section Channel Model

Determination of Flow Parmeters. Both Goldsboro and Kinston are main gauging stations on the Neuse River. One might think it reasonable to take the cross-sectional areas at the measuring stations as representative of the channel. Unfortunately, these gauging stations are generally located where the river has relatively narrow cross sections, so that these areas do not represent a mean cross section over the entire reach.

A net point spacing of 3 mi on the x-axis was chosen, and average cross-sectional areas and average widths as functions of elevation for every 3 mi subreach were determined from the U.S. Geological Survey 15 min series topographic maps (1902, 1942, 1957a,b, 1958b).

A least squares fitting program was used to fit each individual area and top width to polynomial equations of orders 2 through 7. A fourth-order polynomial equation was found to provide the best fit for all 16 areas. The widths B were then easily calculated from the equation $B = dA/dy$. The equation for all 16 cross-sectional areas is given in the following form:

$$A(y,I) = a_4y^4 + a_3y^3 + a_2y^2 + a_1y + a_0,$$

where I is an index referring to the channel location. The coefficients of this equation are tabulated in Table 1. Typical cross sections are shown in Fig. 18.

Channel Bottom Slope S. Channel bottom slope S was calculated between every two neighboring points by the formula,

$$S_i = \frac{\text{Datum height at point i - datum height at point (i+1)}}{\text{Distance between points i and (i+1)}},$$

Datum height is defined as the vertical distance in feet from mean sea level to the channel bottom. These datum heights were measured from the topographic maps based on the contour lines and the two known gauging stations' datum heights.

Results of Computations

The results of the computations for predicting the hydrographs at Kinston for the 1945 and 1964 floods by the implicit

Table 1. Polynomial coefficients of equations for individual cross-sectional areas

I	a_4	a_3	a_2	a_1	a_0
1	0.151655	- 3.04	44.8781	- 0.767774	53.3734
2	0.00517641	3.261238	- 40.2892	300.98	- 29.4285
3	0.00333463	4.06992	- 66.3763	439.725	- 90.4865
4	0.128608	- 1.54256	0.694432	287.073	- 76.2072
5	0.190509	- 5.32675	56.4811	31.2948	12.8542
6	0.113456	- 3.26559	51.6057	25.3572	41.2548
7	-0.00235618	1.02571	13.6706	186.127	3.95523
8	-0.00342456	2.24394	- 6.23079	337.834	- 59.6666
9	-0.039699	3.38345	- 11.632	418.848	-102.304
10	0.0565736	- 1.90667	80.8372	101.179	14.7159
11	-0.272087	15.5952	-186.137	831.848	-158.55
12	-0.0186912	3.96215	- 57.7207	429.661	- 76.3123
13	0.0507101	- 0.00209352	- 9.14404	276.019	- 38.9704
14	0.0507101	- 1.75479	63.7818	99.292	31.1864
15	-0.155109	9.20251	- 89.3087	452.143	- 37.2568
16	-0.129108	9.22434	-117.275	625.933	-109.43

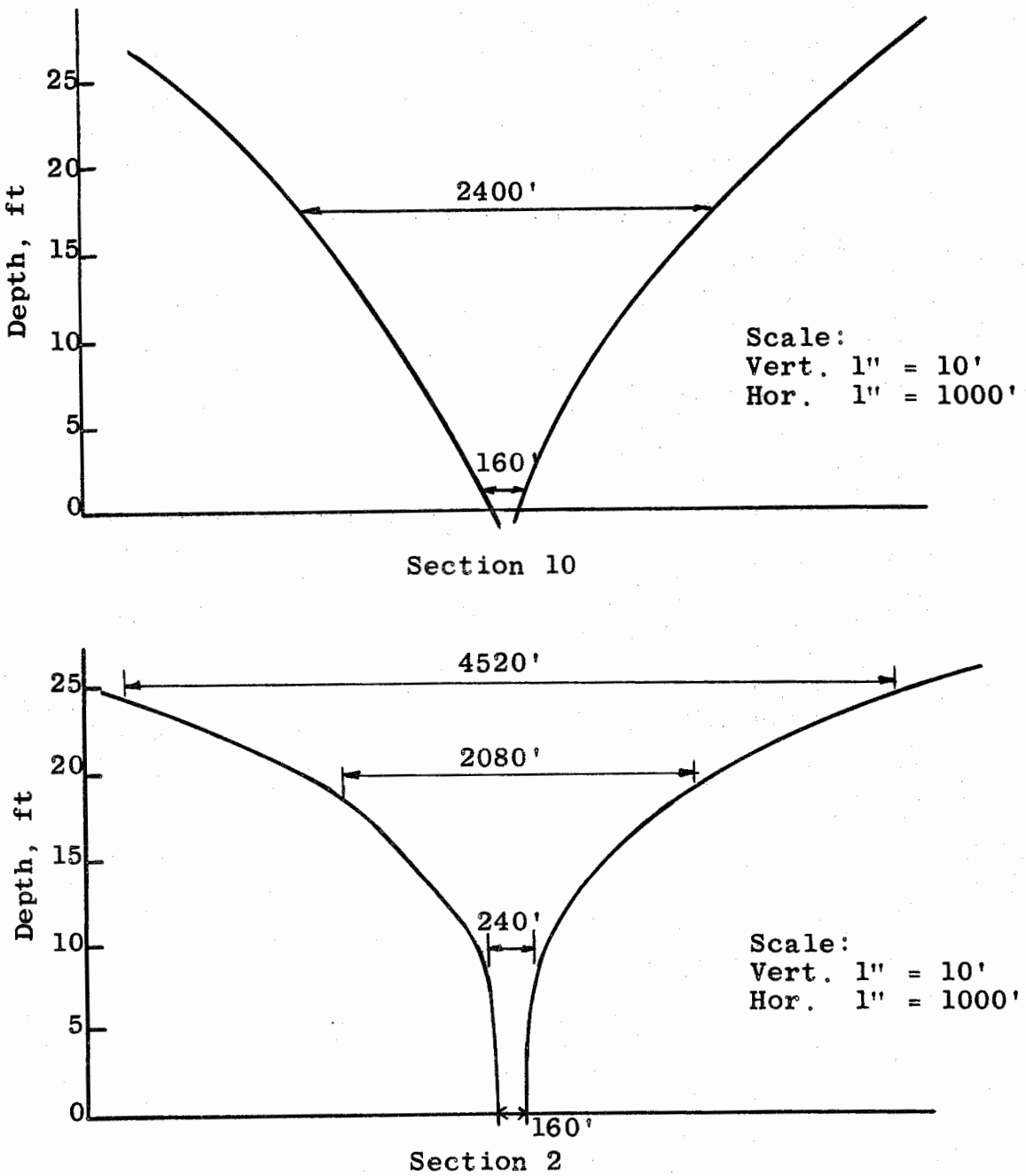


Fig. 18. Typical channel sections taken from small-scale topographic maps

method are presented in Figs. 19 and 20. In these figures, the observed streamflow hydrograph at Kinston is indicated by a solid line and the computed hydrograph by the dashed lines.

Computation Times

Total computer times required to perform the calculations for the major flood of 1945 on the Neuse River, North Carolina, between Goldsboro and Kinston, on an IBM-System 360/75 computer are shown in Table 2. Only 10% more time was required to complete the calculations for predicting the 19-day flood of 1964. Computer times for calculating flows through an idealized channel are not listed.

Table 2. Computer times on an IBM-System 360/75 for calculating flood flows

Method	Channel section	Δt hr	Δx mi	Computer time
Explicit	Composite	0.025	1.5	7 min 49 sec
Explicit	Composite	0.025	3.0	3 min 59 sec
Characteristic	Composite	0.48	2.0	2 min 10 sec
Characteristic	Composite	0.74	4.7	1 min 28 sec
Characteristic	Composite	1.0	7.7	0 min 39 sec
Implicit	Composite	1.0	3.0	1 min 29 sec
Implicit	Composite	5.0	3.0	0 min 55 sec
Implicit	Composite	10.0	3.0	0 min 29 sec
Implicit	Composite	20.0	3.0	0 min 17 sec
Implicit	Discrete	1.0	3.0	1 min 51 sec
Implicit	Discrete	2.0	3.0	1 min 27 sec
Implicit	Discrete	5.0	3.0	0 min 58 sec

The computer times reveal the efficiency of the implicit method over the other methods, particularly over the explicit method. It is also seen that results can be obtained by the implicit method even when such values as $\Delta t = 20$ hr are used for the time step. However, large Δt values may not produce accurate results in many cases because significant flow changes may take place over such an interval.

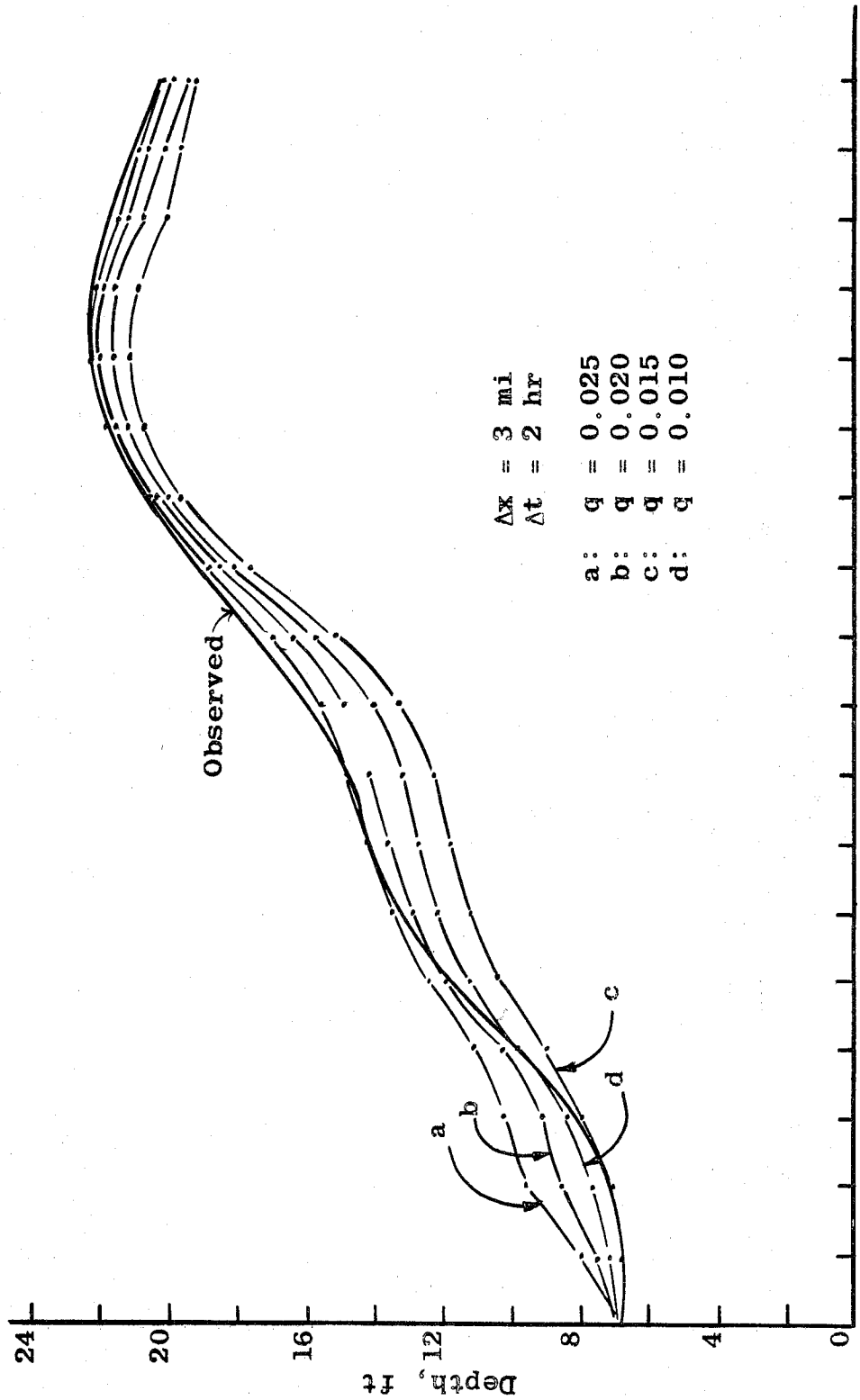


Fig. 19. Calculated 1945 flood hydrograph near Kinston, North Carolina, by the implicit method (discrete section)

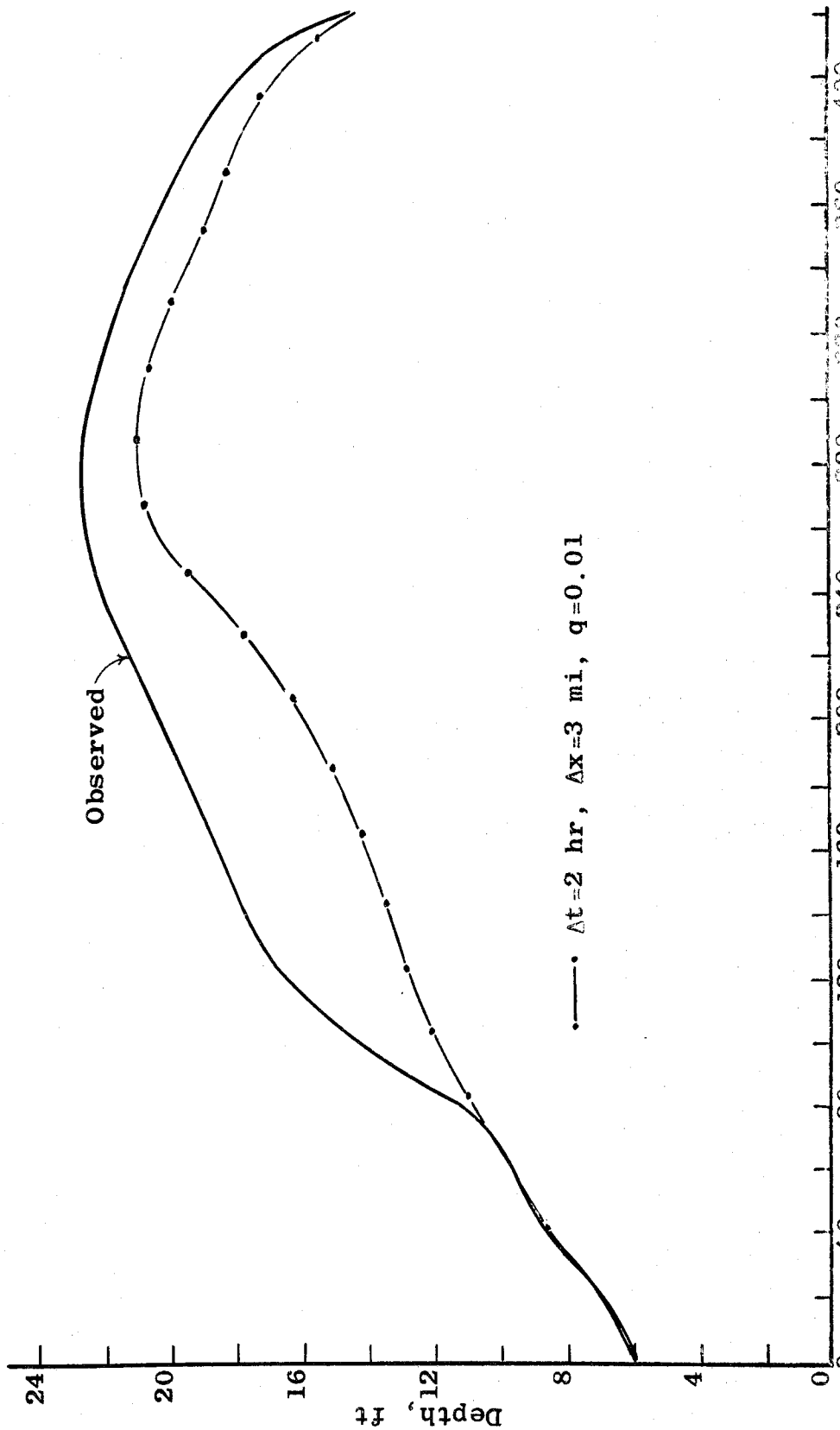


Fig. 20. Calculated 1964 flood hydrograph near Kinston, North Carolina, by the implicit method (discrete section)

Discussion of Results

The results obtained for the flood hydrographs for the 1945 and 1964 floods by the numerical solution of the equations of unsteady flow are in reasonable agreement with the observed values. The maximum deviations between the calculated and observed values of the stage are less than 2 ft and only less than 1 ft at the flood crests. The flow parameters consisting of cross-sectional area A, top width B, lateral flow q, channel bottom slope S, and Manning's coefficient of friction n are decisive factors determining the solutions of the equations of unsteady flow. The channel parameters were determined from small-scale (1:62,500) photographic maps, and no reliable guide for estimating lateral inflow was available. We believe that under the circumstances no better agreement between the observed and computed values could be expected. The same channel sections were used for the 1964 flood as for the flood of 1945. The greater discrepancy between the computed and observed values for the 1964 flood, in our view, reflects the changes in the channel features during the intervening years. We believe that an accurate hydrological survey will provide much better agreement between the observed and computed values.

The computed values using the composite channel section and the discrete section models are not identical. Again, due to inadequate topographical information, meaningful conclusions cannot be derived because the results are influenced by errors in survey data. The solution of problems using a composite section are considerably simplified provided the variations from section to section are small and a single section can realistically represent the channel reach. The discrete section model, however, is far more versatile and has great potentialities. It can be adapted to accept field survey data directly with little reduction.

VIII. SUMMARY AND CONCLUSIONS

The numerical solution of the equations of unsteady flow has been made feasible by the advent of the digital computer. There is little reason to use approximate methods of dubious validity. The complete equations can be integrated for application to a variety of problems.

Three numerical methods--the characteristic, the explicit, and the implicit--were applied to an idealized channel of simple geometry and also to an irregular natural channel. The first application showed that all three methods would give the same results provided small time steps and distance steps are used. In the method of characteristics and in the implicit method, the use of large steps causes errors in the results, the magnitude of the error increasing with the size of the step. However, in both methods the use of smaller time steps than necessary does not improve the accuracy of the results; it only extends computing times. In the explicit method, not only does the accuracy of the results suffer from the use of large time steps, but the procedure becomes unstable and meaningful results cannot be obtained.

The finite difference equations for the explicit method are the simplest of all three, except for the boundary equations where a different approach similar to the method of characteristics has to be used. The explicit method is convenient in that results are given at fixed locations and times. The method can treat cases in which the channel geometry can vary from section to section. However, this variation must be small and there should be very small changes in the bottom slope, otherwise the method fails. Instances of failure of the method, probably due to this, have been reported in the literature. This method is also the least efficient, and because of this is not suitable for routine flood routing where flows extending over many days have to be investigated.

The method of characteristics involves more sophisticated mathematical analysis than the explicit method. It is not suitable for application to natural channels because the times and locations at which results are given are not fixed ahead of time. The method can handle varying channel geometry provided the changes are gradual from section to section. Time steps several multiples of those for the explicit method can be used and still accurate and stable results can be obtained. However, if the method is modified so that through interpolation the results are given at fixed times and locations, the efficiency of the method will be reduced to a considerable extent, and it may be no better than the explicit method. Although this method is an excellent tool for the study of numerous unsteady flow problems, its use in routine fashion for the treatment of unsteady flows in natural channels is questionable.

Mathematically, the implicit method is the most complicated of all three. Results are obtained by the solution of systems of nonlinear equations, and would ordinarily require a great deal of computation. Two factors, however, work in its favor. First, the method is stable for large steps; thus, time steps many times larger than the explicit method, with no accompanying loss of accuracy, can be used. Second, although the system of equations may contain a large number of unknowns, each equation by itself contains at most only four unknowns. By taking advantage of these two factors, an efficient computing procedure was devised, and the method was found to be much faster than the other two. The implicit method also can handle varying channel geometry even where the changes from section to section in the bottom slope are significant. On the basis of the present study, the implicit method should play an important role in flood routing.

It is recommended that great effort should be exerted in applying numerical integration of the equations to rivers, estuaries, and reservoirs for which adequate flow and topographical data are available. Then, a variety of boundary conditions and flow situations can be handled. In conjunction with this effort, computer programs requiring a minimum amount of theoretical knowledge should be written and made available to the average user. It is hoped that in the near future such programs can be used in a routine manner by workers in the field of water resources.

REFERENCES

Abbott, M. B., and F. Ionescu. 1967. On the numerical computation of nearly horizontal flows, J. Hydraulic Research (Netherlands), 5(2), 98-117.

Amein, M. 1966. Streamflow routing on computer by characteristics, Water Resources Research, 2(1), 123-130.

Amein, M. 1967. Some recent studies on numerical flood routing, Proc. 3rd Annual Conf., Am. Water Resources Assoc., San Francisco, California, pp. 274-284.

Amein, M. 1968. An implicit method for numerical flood routing, Water Resources Research, 4(4), 719-726.

Baltzer, R. A., and C. Lai. 1968. Computer simulations of unsteady flows in waterways, J. Hydraulic Div., Proc. ASCE, 94(HY4), 1083-1117.

Brakensiek, D. L., and G. H. Comer. 1965. A re-examination of a flood routing method comparison, J. Hydrology, 1 (3), 223-230.

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

Courant, R., and K. O. Friedrichs. 1948. Supersonic Flow and Shock Waves, Interscience Publishers, Inc., New York.

Crandall, S. H. 1956. Engineering Analysis, McGraw-Hill Book Company, Inc., New York.

Fletcher, A. G., and W. S. Hamilton. 1967. Flood routing in an irregular channel, J. Eng. Mech. Div., Proc. ASCE, 93(EM3), 45-62.

Gilcrest, B. R. 1950. Flood routing, in Engineering Hydraulics, edited by H. Rouse, pp. 635-710, John Wiley, New York.

Henderson, F. M. 1966. Open Channel Flow, Macmillan Company, New York.

Isaacson, E., J. J. Stoker, and B. A. Troesch. 1956. Numerical solution of flood prediction and river regulation problems, Inst. Math. Sci. Rept. No. IMM-235, New York University, New York.

Lighthill, M. J., and G. B. Whitham. 1955. On kinematic waves, Proc. Royal Soc. (London), A229, 281-316.

Liggett, J. A., and D. A. Woolhiser. 1967. Difference solutions of the shallow-water equation, J. Eng. Mech. Div., Proc. ASCE, 9B(EM2), 39-71.

Linsley, R. K., M. A. Kohler, and J. L. H. Paulhus. 1949. Applied Hydrology, McGraw-Hill Book Company, Inc., New York.

Ralston, A. 1965. A First Course in Numerical Analysis, McGraw-Hill Book Company, Inc., New York.

Stoker, J. J. 1957. Water Waves, Interscience Publishers, New York.

Thomas, H. A. 1934. Hydraulics of Flood Movements in Rivers, Carnegie Institute of Technology, Pittsburgh, Pennsylvania.

APPENDIX

The computer programs prepared as part of this study are essentially research tools and do not have the advantage of the sophisticated technique employed by professional programmers. The prime objective of the work was to investigate the feasibility of the various numerical methods. No attempt at preparing library programs for general office work was made.

It should be emphasized that although the same basic principles and methodology are used in nearly all studies of unsteady flows, each particular problem has its own special requirements so far as handling input data and boundary and initial conditions are concerned. If the boundary conditions are different from the type of problems studied in this report, considerable modifications to the programs might be necessary.

FLOOD ROUTING IN THE NATURAL CHANNEL

BY THE EXPLICIT METHOD

```
C                              STREAMFLOW  ROUTING
C                              STOKERS  METHOD
C     Z(K) =CHANNEL  BOTTOM HEIGHT
C     DIST(K)=VALUE OF DISTANCE FOR EACH GRID POINT FROM X=0, IN MILES
C     SO   =CHANNEL  BOTTOM SLOPE
C     T    =TIM AT ANY INSTANT
C     N    =NUMBER OF POINTS GIVEN ON T-AXIS
C      CI(Y)=A FUNCTION OF DEPTH OF MANNING'S ROUGHNESS COEFFICIENT AT
C       UPSTREAM END
C      ENDIS(I)=VALUE OF DISTANCE IN THE SECOND ROW,IN MILES
C      ENVEL(I)=VALUE OF VELOCITY IN THE SECOND ROW,IN FT/SEC
C      ENDEL(I)=VALUE OF STAGE    IN THE SECOND ROW,IN FT
C      ENDEP(I)=VALUE OF DEPTH  ON   THE SECOND ROW,IN FT
C      BEDIS(I)=VALUE OF DISTANCE ON X-AXIS ANY ADD ROW IN MILES
C      TIME     =TIME  AT ANY  INSTANT    IN  HOURS
C      BEVEL(I)=VALUES OF  VELOCITY ON ADD ROW  IN  FT/SEC
C      BEDEP(I)=VALUES OF DEPTH    ON ADD ROW  IN  FT
C      BEGEL(I)=VALUES OF  STAGE    ON ADD ROW  IN  FT
C      QZERO =INITIAL VALUE OF CELERITY  ON THE T-AXIS
C      M      =NUMBER OF POINTS GIVEN ON THE T-AXIS
       COMMON    STAGE(200),BEVEL(100),BEDIS(100),BEGEL(100),BEDEP(100) ,
      1DISCH(100),Z(100),ENVEL(100),ENDIS(100),ENDEL(100),T(200),Q(100) ,
      2QG(100),ENDEP(100),DISHI,DISHN,XFIN,ELEVN,DN,TZERO,VL1,TFIN
       COMMON IP,N,M,L,DELX,DELT,SC,S,TIME,NN,V1,V2,DT,DX,G,H1,H2,SF1,Q1
      1,Q2,A1,A2,ELEVO,XZERO,R1,R2,D1,SF2,EH1,R1,VM,HM,HL,J,DL1,CP,V,I,R2
      2,FA,DL2,DV,DY,LL,VP,DP,HP,DTIME,DM,AL1,AL2,EL1,CL,BL1,BL2,DL,VL,D2
C      WETTED PERIMETER IS EQUAL TO THE TOP WIDTH
    2  FORMAT(16I5)
    3 FORMAT(8E10.3)
       IP=10
       READ(1,2) N,M,L
       READ(1,3) TFIN,DTIME,TZERO,XZERO,XFIN,ELEVO,ELEVN,G,DN,DISHI,DISHN
      1 ,DELX,DELT
       READ(1,3)(STAGE(K),K=1,M)
       CALL EXPL1
  100  CALL EXPL2
       CALL EXPL3
       CALL EXPL4
       CALL EXPL5
       CALL EXPL6
       CALL EXPL11
       CALL EXPL12
  255  IF(TIME-TFIN) 100,260,260
  260  CALL EXIT
       END
       SUBROUTINE EXPL1
       COMMON    STAGE(200),BEVEL(100),BEDIS(100),BEGEL(100),BEDEP(100) ,
      1DISCH(100),Z(100),ENVEL(100),ENDIS(100),ENDEL(100),T(200),Q(100) ,
      2QG(100),ENDEP(100),DISHI,DISHN,XFIN,ELEVN,DN,TZERO,VL1,TFIN
       COMMON IP,N,M,L,DELX,DELT,SC,S,TIME,NN,V1,V2,DT,DX,G,H1,H2,SF1,Q1
      1,Q2,A1,A2,ELEVO,XZERO,R1,R2,D1,SF2,EH1,R1,VM,HM,HL,J,DL1,CP,V,I,R2
      2,FA,DL2,DV,DY,LL,VP,DP,HP,DTIME,DM,AL1,AL2,EL1,CL,BL1,BL2,DL,VL,D2
    4 FORMAT(///45X,19HSTREAMFLOW ROUTING)
       WRITE(3,4)
       DI=STAGE(1)
```

```
      V1=DISHI/A(DI)
      VN=DISHN/A(DN)
      SO=(DISHN-DISHI)/(XEIN*5280.)
      S=(ELEVO-ELEVN)/(XEIN*5280.)
   30 BEDIS(1)=XZERO
      TIME=TZERO
      T(1)=TZERO
      Z(1)=ELEVO
      NN=N-1
      MM=M-1
      DT=DELT*3600.0
      DX=DELX*5280.0
      DO 33 K=1,M
   33 STAGE(K)=STAGE(K)+ELEVO
      DO 50 I=1,NN
      BEDIS(I+1)=BEDIS(I)+DELX
   50 Z(I+1)=ELEVO-(BEDIS(I+1)-XZERO)*S*5280.0
      DO 60 I=1,N
      DG(I)=(BEDIS(I)-XZERO)/(XEIN-XZERO)
      BEDEP(I)=DI+(DN-DI)*DG(I)
      BEGEL(I)=BEDEP(I)+Z(I)
      DISCH(I)=DISHI+(DISHN-DISHI)*DG(I)
      BEVEL(I)=V1+(VN-V1)*DG(I)
   60 Q(I)=SO
C     *       *       *       *       *       *       *
C     THE INITIAL VALUES ON THE X-AIXS ARE PRESENTED AS FOLLOWS
C     *       *       *       *       *       *       *
      DO 70 K=1,MM
   70 T(K+1)=T(K)+DTIME
      RETURN
      END
      SUBROUTINE EXPL2
C     *       *       *       *       *       *       *
C     NOTE THE ROWS ARE NUMBERED STARTING WITH UNITY ON THE ROW WITH THE
C    1INITIAL TIME
C     COMPUTATION OF THE AUXILIARY POINT AT THE DOWNSTREAM END OF    THE
C    1EVEN ROW
C     *       *       *       *       *       *       *
      COMMON    STAGE(200),BEVEL(100),BEDIS(100),BEGEL(100),BEDEP(100) ,
     1DISCH(100),Z(100),ENVEL(100),ENDIS(100),ENDEL(100),T(200),Q(100) ,
     2DG(100),ENDEP(100),DISHI,DISHN,XEIN,ELEVN,DN,TZERO,VI1,TEIN
      COMMON ID,L,M,I,DELX,DELT,SO,S,TIME,NN,V1,V2,DT,DX,G,H1,H2,SF1,01
     1,02,A1,A2,ELEVO,XZERO,B1,B2,D1,SF2,EH1,R1,VM,HM,HL,J,DL1,CP,V,I,R2
     2,EA,DL2,DV,DY,II,VP,DP,EP,DTIME,DM,AL1,AL2,EL1,CL,BL1,BL2,DL,VL,D2
C     *       *       *       *       *       *       *
C     CALCULATION OF THE VARIBLES AT THE INTERIOR POINTS ON EVEN ROWS
C     *       *       *       *       *       *       *
      TIME=TIME+DELT
      DO 120 I=1,NN
      CALL EXPL2
      ENDIS(I)=BEDIS(I)+DELX/2.0
      ENVEL(I)=.5*(V1+V2)-(DT/DX)*(.5*(V2*V2-V1*V1)+G*(H2-H1))-G*DT*.5*
     1(SF1+SF2)-.5*DT*(V2/A2+V1/A1)*Q(I)
      Z(I)=ELEVO-(ENDIS(I)-XZERO)*S*5280.0
      ENDEL(I)=.5*(H1+H2)+(2.*DT/(B1+B2))*(Q(I)+(A1*V1-A2*V2)/DX)
      ENDEP(I)=ENDEL(I)-Z(I)
```

```
  120 DISCH(I)=ENVEL(I)*ENDEP(I)*A1/BEDEP(I)
      RETURN
      END
      SUBROUTINE  EXPL3
C     *       *       *       *       *       *       *
C     COMPULATION OF THE AUXILIARY POINT ON THE T-AXIS ON AN EVEN ROW
C     *       *       *       *       *       *       *
      COMMON    STAGE(200),BEVEL(100),BEDIS(100),BEGEL(100),BEDEP(100) ,
     1DISCH(100),Z(100),ENVEL(100),ENDIS(100),ENDEL(100),T(200),Q(100) ,
     2QG(100),ENDEP(100),DISHI,DISHN,XFIN,ELEVN,DN,TZERO,VL1,TFIN
      COMMON IP,N,M,L,DELX,DELT,SQ,S,TIME,NN,V1,V2,DT,DX,G,H1,H2,SF1,Q1
     1,Q2,A1,A2,ELEVQ,XZERO,R1,R2,D1,SF2,EH1,R1,VM,HM,HL,J,DL1,CP,V,I,R2
     2,EA,DL2,DV,DY,LL,VP,DP,HP,DTIME,DM,AL1,AL2,EL1,CL,BL1,BL2,DL,VL,D2
C     *       *       *       *       *       *       *
C     COMPULATION OF VARIABLE AT THE DOWNSTREAM END OF AN ODD ROW
C     *       *       *       *       *       *       *
    9 FORMAT(1H I4,6(5H     F10.3))
      I=NN
      LL=0
      CALL EXPL7
      CP=SQRT(A2*R2/G)
      V=V2
  121 EA=A(D2)
      EH1=D2+Z(N)
      R2=R2*(EH1-H2)/DT+(A2*V2-A1*V1)/DX-Q1+CP*((V-V2)/DT+(V2*V2-V1*V1
     1)/(2.*DX)+G*(H2-H1)/DX+Q1*V2/A2+G*SF2)
      CALL EXPL8
      VP=V-DV
      DP=D2-DY
      IF(LL-19) 122,122,126
  122 LL=LL+1
      IF(ABS(V-VP)-0.01) 123,124,124
  123 IF(ABS(DP-D2)-0.01) 127,124,124
  124 V=VP
      D2=DP
      GO TO 121
  126 WRITE(3,9)N, VP,DP,DV,DY
  127 HP=Z(N)+DP
      ENDLP=HP
      DISHP=VP*DP*A2/BEDEP(N)
      RETURN
      END
      SUBROUTINE  EXPL4
C     *       *       *       *       *       *       *
C     COMPULATION OF VARIABLE ON AN ODD ROW
C     *       *       *       *       *       *       *
      COMMON    STAGE(200),BEVEL(100),BEDIS(100),BEGEL(100),BEDEP(100) ,
     1DISCH(100),Z(100),ENVEL(100),ENDIS(100),ENDEL(100),T(200),Q(100) ,
     2QG(100),ENDEP(100),DISHI,DISHN,XFIN,ELEVN,DN,TZERO,VL1,TFIN
      COMMON IP,N,M,L,DELX,DELT,SQ,S,TIME,NN,V1,V2,DT,DX,G,H1,H2,SF1,Q1
     1,Q2,A1,A2,ELEVQ,XZERO,R1,R2,D1,SF2,EH1,R1,VM,HM,HL,J,DL1,CP,V,I,R2
     2,EA,DL2,DV,DY,LL,VP,DP,HP,DTIME,DM,AL1,AL2,EL1,CL,BL1,BL2,DL,VL,D2
      IF(TIME-T(L)) 134,131,132
  131 SUREL=STAGE(L)
      GO TO 136
  132 L=L+1
```

```
134 SUREL=STAGE(L-1)+(TIME-T(L-1))*(STAGE(L)-STAGE(L-1))/DTIME
134 DL=SUREL-FLEVO
    HL=SUREL
    I=1
    CALL EXPL7
    CL=SQRT(G/(B1*A1))
    VM=ENVEL(1)
    DM=ENDEP(1)
    HM=ENDEL(1)
    VL=V1-DT*((V2*V2-V1*V1+G*2.*(H2-H1))/(2.*DX)+C1*V1/A1+G*SF1)+CL
   1*2.*(HL-H1)+DT*CL*(A2*V2-A1*V1)/DX-QL*DT*CL
    DISCH(1)=VL*DL*A1/BEDEP(1)
    RETURN
    END
    SUBROUTINE EXPL5
    COMMON    STAGE(200),BEVEL(100),BEDIS(100),BEGEL(100),BEDEP(100) ,
   1DISCH(100),Z(100),ENVEL(100),ENDIS(100),ENDEL(100),T(200),Q(100) ,
   2DG(100),ENDEP(100),DISHI,DISHN,XFIN,FLEVN,DN,TZERO,VL1,TFIN
    COMMON IP,N,M,L,DELX,DELT,SC,S,TIME,NN,V1,V2,DT,DX,G,H1,H2,SF1,Q1
   1,Q2,A1,A2,FLEVO,XZERO,B1,B2,D1,SF2,EH1,R1,VM,HM,HL,J,DL1,CP,V,I,R2
   2,FA,DL2,DV,DY,LL,VP,QP,HP,DTIME,DM,AL1,AL2,FL1,CL,BL1,BL2,DL,VL,D2
    DO 140 I=1,NN
    BEDIS(I)=ENDIS(I)
    BEGEL(I)=ENDEL(I)
    BEVEL(I)=ENVEL(I)
140 BEDEP(I)=ENDEP(I)
200 TIME=TIME+DELT
    IF(TIME-T(L))210,207,208
207 ENDEL(1)=STAGE(L)
    GO TO 212
208 L=L+1
210 ENDEL(1)=STAGE(L-1)+(TIME-T(L-1))*(STAGE(L)-STAGE(L-1))/DTIME
212 Z(1)=FLEVO
    ENDIS(1)=0.
    ENDEP(1)=ENDEL(1)-Z(1)
    QL=Q(L)
    CL1=CL
    CL2=CM
    CALL EXPL10
    ENVEL(1)=VL-DT*((VM*VM-VL*VL+G*2.*(HM-HL))/DX+QL*VL/AL1+EL1)+CL*
   1BL1*(ENDEL(1)-HL)+DT*CL*(AL2*VM-AL1*VL)/(.5*DX)-QL*DT*CL
    DISCH(1)=ENVEL(1)*ENDEP(1)*AL1/DL
    RETURN
    END
    SUBROUTINE EXPL6
    COMMON    STAGE(200),BEVEL(100),BEDIS(100),BEGEL(100),BEDEP(100) ,
   1DISCH(100),Z(100),ENVEL(100),ENDIS(100),ENDEL(100),T(200),Q(100) ,
   2DG(100),ENDEP(100),DISHI,DISHN,XFIN,FLEVN,DN,TZERO,VL1,TFIN
    COMMON IP,N,M,L,DELX,DELT,SC,S,TIME,NN,V1,V2,DT,DX,G,H1,H2,SF1,Q1
   1,Q2,A1,A2,FLEVO,XZERO,B1,B2,D1,SF2,EH1,R1,VM,HM,HL,J,DL1,CP,V,I,R2
   2,FA,DL2,DV,DY,LL,VP,QP,HP,DTIME,DM,AL1,AL2,FL1,CL,BL1,BL2,DL,VL,D2
    J=NN-1
    DO 220 I=1,J
    CALL EXPL7
    ENDIS(I+1)=BEDIS(I)+DELX/2.0
    ENVEL(I+1)=.5*(V1+V2)-(DT/DX)*(.5*(V2*V2-V1*V1)+G*(H2-H1))-G*DT*
```

```
     1.5*(SF1+SF2)-.5*DT*(V2/A2+V1/A1)*Q(I+1)
      Z(I+1)=ELEVO-(ENDIS(I+1)-XZERO)*S*5280.0
      ENDEL(I+1)=.5*(H1+H2)+(2.*DT/(B1+B2))*(Q(I)+(A1*V1-A2*V2)/DX)
      ENDEP(I+1)=ENDEL(I+1)-Z(I+1)
  220 DISCH(I+1)=ENVEL(I+1)*ENDEP(I+1)*A2/BEDEP(I+1)
      RETURN
      END
      SUBROUTINE  EXPL7
      COMMON    STAGE(200),BEVEL(100),BEDIS(100),BEGEL(100),BEDEP(100) ,
     1DISCH(100),Z(100),ENVEL(100),ENDIS(100),ENDEL(100),T(200),Q(100) ,
     2DG(100),ENDEP(100),DISHI,DISHN,XFIN,ELEVN,DN,TZERO,VL1,TFIN
      COMMON IP,N,M,L,DELX,DELT,SQ,S,TIME,NN,V1,V2,DT,DX,G,H1,H2,SF1,Q1
     1,Q2,A1,A2,ELEVO,XZERO,B1,B2,D1,SF2,EH1,R1,VM,HM,HL,J,DL1,CP,V,I,R2
     2,FA,DL2,DV,DY,LL,VP,DP,HP,DTIME,DM,AL1,AL2,EL1,CL,BL1,BL2,DL,VL,D2
      V1=BEVEL(I)
      V2=BEVEL(I+1)
      Q1=Q(I)
      Q2=Q(I+1)
      H1=BEGEL(I)
      H2=BEGEL(I+1)
      D1=BEDEP(I)
      D2=BEDEP(I+1)
      C1=CI(D1)
      C2=CI(D2)
      C3=CN(D1)
      C4=CN(D2)
      CD1=C1+(C3-C1)*DG(I)
      CD2=C2+(C4-C2)*DG(I+1)
      A1=A(D1)
      A2=A(D2)
      P1=P(D1)
      P2=P(D2)
      B1=DAY(D1)
      B2=DAY(D2)
      R1=(P1/A1)**(4./3.)
      R2=(P2/A2)**(4./3.)
      SF1=CD1*CD1*V1*V1*R1/2.2082
      SF2=CD2*CD2*V2*V2*R2/2.2082
      RETURN
      END
      SUBROUTINE  EXPL9
      COMMON    STAGE(200),BEVEL(100),BEDIS(100),BEGEL(100),BEDEP(100) ,
     1DISCH(100),Z(100),ENVEL(100),ENDIS(100),ENDEL(100),T(200),Q(100) ,
     2DG(100),ENDEP(100),DISHI,DISHN,XFIN,ELEVN,DN,TZERO,VL1,TFIN
      COMMON IP,N,M,L,DELX,DELT,SQ,S,TIME,NN,V1,V2,DT,DX,G,H1,H2,SF1,Q1
     1,Q2,A1,A2,ELEVO,XZERO,B1,B2,D1,SF2,EH1,R1,VM,HM,HL,J,DL1,CP,V,I,R2
     2,FA,DL2,DV,DY,LL,VP,DP,HP,DTIME,DM,AL1,AL2,EL1,CL,BL1,BL2,DL,VL,D2
      EQ=BQ(D2)
      R1=FA*V-EQ
      DADY=DAY(D2)
      DQDY=DQ(D2)
      DR1DV=FA
      DR1DY=V*DADY-DQDY
      DR2DV=CP/DT
      DR2DY=B2/DT
      DATA=DR1DV*DR2DY-DR1DY*DR2DV
```

```
      DA1=R1*DR2DV-R2*DR1DY
      DA2=R2*DR1DV-R1*DR2DV
      DV=DA1/DATA
      DY=DA2/DATA
      RETURN
      END
      SUBROUTINE  EXPL10
      COMMON    STAGE(200),BEVEL(100),BEDIS(100),BEGEL(100),BEDEP(100) ,
     1DISCH(100),Z(100),ENVEL(100),ENDIS(100),ENDEL(100),T(200),Q(100) ,
     2DG(100),ENDEP(100),DISHI,DISHN,XFIN,ELEVN,DN,TZERO,VL1,TFIN
      COMMON IP,N,M,L,DELX,DELT,SQ,S,TIME,NN,V1,V2,DT,DX,G,H1,H2,SF1,Q1
     1,Q2,A1,A2,ELEVO,XZERO,R1,R2,D1,SF2,EH1,R1,VM,HM,HL,J,DL1,CP,V,I,R2
     2,EA,DL2,DV,DY,LL,VP,DP,HP,DTIME,DM,AL1,AL2,EL1,CL,BL1,BL2,DL,VL,D2
  137 AL1=A(DL1)
      AL2=A(DL2)
      BL1=DAY(DL1)
      BL2=DAY(DL2)
      PL1=R(DL1)
      PL2=R(DL2)
      CL1=CI(DL1)
      CL=SQRT(G/(PL1*AL1))
      HRR1=(PL1/AL1)**(4./3.)
      EL1=G*CL1*CL1*VL*VL*HRR1/2.2032
      RETURN
      END
      SUBROUTINE  EXPL11
      COMMON    STAGE(200),BEVEL(100),BEDIS(100),BEGEL(100),BEDEP(100) ,
     1DISCH(100),Z(100),ENVEL(100),ENDIS(100),ENDEL(100),T(200),Q(100) ,
     2DG(100),ENDEP(100),DISHI,DISHN,XFIN,ELEVN,DN,TZERO,VL1,TFIN
      COMMON IP,N,M,L,DELX,DELT,SQ,S,TIME,NN,V1,V2,DT,DX,G,H1,H2,SF1,Q1
     1,Q2,A1,A2,ELEVO,XZERO,R1,R2,D1,SF2,EH1,R1,VM,HM,HL,J,DL1,CP,V,I,R2
     2,EA,DL2,DV,DY,LL,VP,DP,HP,DTIME,DM,AL1,AL2,EL1,CL,BL1,BL2,DL,VL,D2
    9 FORMAT(1H I4,6(5H     F10.3))
      ENDIS(N)=ENDIS(NN)+DELX
      Z(N)=ELEVO-(ENDIS(N)-XZERO)*S*5280.
      I=NN
      LL=0
      BEDEP(N)=DP
      BEGEL(N)=HP
      CALL EXPL7
      V=V2
      CP=SQRT(A2*R2/G)
  221 EA=A(D2)
      EH1=D2+Z(N)
      R2=R2*(EH1-H2)/DT+(A2*V2-A1*V1)/(.5*DX)-Q1+CP*((V-V2)/DT+(V2*V2-V1
     1*V1)/DX+G*(H2-H1)/(DX*.5)+CI*V2/A2+G*SF2)
      CALL EXPL9
      ENVEL(N)=V-DV
      ENDEP(N)=D2-DY
      IF(LL-17) 222,222,226
  222 LL=LL+1
      IF(ABS(ENVEL(N)-V)-0.01) 223,224,224
  223 IF(ABS(ENDEP(N)-D2)-0.01) 227,224,224
  224 V=ENVEL(N)
      D2=ENDEP(N)
      GO TO 221
```

```
226   WRITE(3,9) N,ENVEL(N),ENDEP(N),DV,DY
  227 ENDEL(N)=Z(N)+ENDEP(N)
      ENVEL(N)=V
      DISCH(N)=ENVEL(N)*ENDEP(N)*A2/DP
      RETURN
      END
      SUBROUTINE  EXPL12
      COMMON    STAGE(200),BEVEL(100),BEDIS(100),BEGEL(100),BEDEP(100) ,
     1DISCH(100),Z(100),ENVEL(100),ENDIS(100),ENDEL(100),T(200),Q(100) ,
     2DG(100),ENDEP(100),DISHI,DISHN,XFIN,ELEVN,DN,TZERO,VL1,TFIN
      COMMON IP,N,M,I,DELX,DELT,SQ,S,TIME,NN,V1,V2,DT,DX,G,H1,H2,SF1,Q1
     1,Q2,A1,A2,ELEVO,XZERO,D1,B2,D1,SF2,FH1,R1,VM,HM,HL,J,DL1,CP,V,I,R2
     2,FA,DL2,DV,DY,LL,VP,DP,BP,DTIME,DM,AL1,AL2,EL1,CL,BL1,BL2,DL,VL,D2
    7 FORMAT(1H ///15X,4HTIME,9X,8HDISTANCE,8X,5HSTAGE,11X,5HDEPTH,10X ,
     18HVELOCITY,5X,9HDISCHARGE)
    8 FORMAT(1H 13X,7H(HOURS),7X,7H(MILES),9X,5H(FT.),11X,5H(FT.),8X   ,
     110H(FT./SEC.),4X,12H(CUFT./SEC.))
    9 FORMAT(1H I4,6(5H     F10.3))
      IF(TIME-IP)231,229,229
  229 WRITE(3,7)
      WRITE(3,8)
      IP=IP+20
      DO 230 I=1,N
  230 WRITE(3,9)I,TIME,ENDIS(I),ENDEL(I),ENDEP(I),ENVEL(I),DISCH(I)
  231 DO 240 I=1,N
      BEDIS(I)=ENDIS(I)
      BEGEL(I)=ENDEL(I)
      BEVEL(I)=ENVEL(I)
  240 BEDEP(I)=ENDEP(I)
      RETURN
      END
      FUNCTION A(Y)
      A=2.20489*Y*Y*Y-9.02071*Y*Y+190.635*Y+228.779
      RETURN
      END
      FUNCTION B(Y)
      B=(((-.0345819*Y+1.74116)*Y-21.1231)*Y+131.725)*Y+28.3223
      RETURN
      END
      FUNCTION BQ(Y)
      BQ=-.324372*Y*Y*Y*Y+21.5435*Y*Y*Y-413.972*Y*Y+3668.52*Y-9335.86
      RETURN
      END
      FUNCTION CI(Y)
      CI=(((((.281877E-06*Y-.24522E-04)*Y+.000762819)*Y-.0102767)*Y+
     10.0614686)*Y-0.0876877
      RETURN
      END
      FUNCTION CN(Y)
      CN=(((.187844E-5*Y-.000119336)*Y+.00264335)*Y-.02147)*Y+.093644
      RETURN
      END
      FUNCTION DAY(Y)
      DAY=6.61467*Y*Y-16.04142*Y+190.635
      RETURN
      END
```

```
      FUNCTION DRY(Y)
      DRY=((-0.1383276*Y+5.223481)*Y-42.2462)*Y+131.725
      RETURN
      END
      FUNCTION DQ(Y)
      DQ=-1.297489*Y*Y*Y+64.6305*Y*Y-829.944*Y+3668.52
      RETURN
      END
```

FLOOD ROUTING IN THE NATURAL CHANNEL

BY THE METHOD OF CHARACTERISTICS

```
C                STREAMFLOW ROUTING ON COMPUTER BY CHARACTERISTICS
C     IN THIS PROGRAM DISTANCE IS GIVEN IN MILES AND TIME IN HOURS
C
C     NOTE  =A REFERENCE NUMBER WHICH IS 1 FOR COMPUTATIONS
C    TERMINATING  ON T-AXIS AND 2 FOR THOSE ON THE FINAL
C      CHARACTERISTIC ISSUED FROM J
C     INDEX =A REFERENCE NUMBER WHOSE VALUES ARE SAME AS NOTE
C     IK =SUBSCRIBED INDEX FOR IDENTIFYING THE ENDING   NUMBER OF
C     THE CALCULATING  CHARACTERISTIC
C     IK =SUBSCRIBED INDEX FOR IDENTIFYING THE STARTING NUMBER OF
C     THE CALCULATING  CHARACTERISTIC
C     ML =AN INDEX FOR IDENTIFYING THE CURRENT VALUE  OF TIME ON
C      T-AXIS
C     UZERO=INITIAL VALUE OF VELOCITY IN THE CHANNEL
C     CO(K) =MANNING'S ROUGHNESS COEFFICIENT AT EACH POINT
C      BASED ON DEPTH
C     EXCEL(K)=VALUES OF CELERITY=C=SQRT(G*Y)  ON THE T-AXIS
C     EXTIM(K)=VALUES OF TIME ON THE T-AXIS, IN  SEC
C    BETIM(K)=VALUES OF TIME ON THE CURVE OF KNOWN CHARACTERISTIC
C     BEDIS(K)=VALUES OF X  ON THE CURVE  OF KNOWN  CHARACTERISTIC
C     BEVEL(K)= VELOCITIESY ON THE CURVE OF KNOWN CHARACTERISTIC
C     BECEL(K)= CELERITIES  ON THE CURVE OF KNOWN CHARACTERISTIC
C     ENTIM(K)=TIMES ON THE CURVE OF UNKNOWN CHARACTERISTIC
C    ENDIS(K)=VALUES OF X ON THE CURVE  OF UNKNOWN CHARACTERISTIC
C     ENVEL(K)=VELOCITIES ON THE CURVE OF UNKNOWN CHARACTERISTIC
C     ENCEL(K)= CELERITIES  ON THE CURVE OF UNKNOWN CHARACTERISTIC
C     BAREA(K)=VALUES OF AREA ON THE CURVE OF KNOWN CHARACTERISTIC
C     EAREA(K)= AREAS  ON THE CURVE OF UNKNOWN CHARACTERISTIC
C     BWIDT(K)= T=DA/DY ON THE CURVE OF KNOWN CHARACTERISTIC
C     EWIDT(K)= T=DA/DY ON THE CURVE OF UNKNOWN CHARACTERISTIC
C     BCO(K)=MANNING'S ROUGHNESS COEFFICIENTS AT KNOWN
C      CHARACTERISTIC
C     ECO(K)=MANNING'S ROUGHNESS COEFFICIENTS  AT UNKNOWN
C      CHARACTERISTIC
C     BDG(K)=INTERPOLATION FORMULA AT KNOWN CHARACTERISTICS
C     EDG(K)=INTERPOLATION FORMULA AT UNKNOWN CHARACTERISTICS
C      J  =AN INDEX FOR IDENTIFYINY THE TIME ALONG T-AXIS
C    A = A FUNCTION OF CROSSSECTIONAL AREAS
C     DAY = A FUNCTION OF THE DERIVATIVE OF A(DAY=DA/DY)
C    CI = MANNING'S ROUGHNESS COEFFICIENT EQUATION AT UPSTREAM END
C     CN=MANNING'S ROUGHNESS COEFFICIENT AT DOWNSTREAM END
C     CI AND CN ARE WRITTEN AS A FUNCTION IN THE PROGRAM
C     B = A FUNCTION OF TOP-WIDTHS
C
C          THIS IS THE MAIN PROGRAM
      COMMON EXCEL(830),EXTIM(830),BETIM(830),BEDIS(830),BDG(800),
     1    TIME(830),BECEL(830),ENTIM(830),ENDIS(830),ENVEL(830),
     2    DIST(830),BAREA(830),EAREA(830),BWIDT(800),EWIDT(830),
     3    DISCH(830),HEIGH(830),BEDEP(800),ENDEP(830),BEVEL(800),
     4    BCO(800),ECO(800),Z(800),STAGE(800),ENCEL(800),EDG(800)
      COMMON J,TDIPH,JN,MJ,DT,V,LL,DL,C,GG,S,K,IK,NOTE,INDEX,ML,
     1    IP,SO,N,JK,XFIN,XZERO,TZERO,DEPI,DEPN,VI,VN
3     FORMAT(16I5)
    4 FORMAT (///30X,'FLOOD ROUTING BY METHOD OF CHARACTERISTICS')
    5 FORMAT (8F10.3)
```

```
    6 FORMAT (///7X,30X,7HINITIAL)
    7 FORMAT (///9X,5HDEPTH,11X,8HDISTANCE,7X,4HTIME,11X,8HVELOCITY
     1,8X,8HCELERITY,4X,9HDISCHARGE,7X,5HSTAGE///)
    8 FORMAT (1H ,I3,7F15.6)
    9 FORMAT (1H0)
      READ(1,3) M,NOTE,J,JK,ML,INDEX,IK
      READ(1,5) ELEVO,ELEVN,DISHT,DISHN,DEPN,G,TEIN,XEIN,TZERO,
     1    XZERO,TOTPH
      READ(1,5)(STAGE(K),K=1,M)
      TEIN=TEIN*3600.
      IP=1
C READ INITIAL VALUES AS INPUT DATA AND THEN PRINT OUT THE STAGE
      GG=SQRT(G)
      S=(ELEVO-ELEVN)/XEIN
      DEPI=STAGE(1)
      SO=0.001
      VI=DISHT/A(DEPI)
      VN=DISHN/A(DEPN)
      UZERO=VI
      DO 20 K=1,M
      DS=STAGE(K)
      BAREA(K)=A(DS)
      BWIDT(K)=B(DS)
   20 EXCEL(K)=SQRT(32.172*BAREA(K)/BWIDT(K))
      CZERO=EXCEL(1)
      EXTIM(1)=TZERO
      BETIM(IK)=TZERO
      BEDIS(IK)=XZERO
      BEDEP(IK)=DEPI
      BEVEL(IK)=UZERO
      BECEL(IK)=CZERO
      DT=0.25*(CZERO-UZERO)*TOTPH/CZERO
      WRITE(3,4)
      WRITE(3,6)
   25 CALL CHAR3
   60 IF(MJ-1) 97,61,61
   61 KMAX=MJ
      IF(INDEX-1) 63,63,62
   62 KMAX=MJ+1
   63 DO 90 K=IK,KMAX
      CALL CHAR4
   90 CONTINUE
C
C     CALCULATING UPSTREAM BOUNDARY POINTS
C
   97 ENDIS(JK)=XZERO
      AS=ENVEL(JN)-EXCEL(JN)
      ENTIM(JK)=ENTIM(JN)+(ENDIS(JK)-ENDIS(JN))/AS
      IF(ENTIM(JK)-EXTIM(ML)) 99,98,98
   98 ML=ML+1
      IP=1
   99 ENCEL(JK)=EXCEL(ML)+(EXCEL(ML)-EXCEL(ML-1))*(ENTIM(JK)-
     1   EXTIM(ML))/(EXTIM(ML)-EXTIM(ML-1))
      ENDEP(JK)=STAGE(ML)+(ENTIM(JK)-EXTIM(ML))*(STAGE(ML)-
     1   STAGE(ML-1))/TOTPH
      CALL CHAR5
```

```
      IF(INDEX-1) 124,124,135
  124 GO TO (131,129),NOTE
  129 INDEX=2
      GO TO 135
  131 IF(ML-M) 135,134,134
  134 CORR=DT*(EXTIM(ML)-ENTIM(JK))/(ENTIM(JK)-BETIM(JK-1))
      ENTIM(IK)=ENTIM(IK)+CORR
      NOTE=2
      GO TO 206
  135 ENDEP(JK)=STAGE(ML)+(ENTIM(JK)-EXTIM(ML))*(STAGE(ML)-STAGE(ML-1))/
     1TDIPH
140   IF(IP-1) 141,141,170
  141 DO 142 K=IK,JK
      Z(K)=ELEVO-(DIST(K)-XZERO)*S*5280.
      HEIGH(K)=ENDEP(K)+Z(K)
  142 DISCH(K)=EAREA(K)*ENVEL(K)
      WRITE(3,7)
      DO 150 K=IK,JK
  150 WRITE(3,8)K,ENDEP(K),DIST(K),TIME(K),ENVEL(K),ENCEL(K),DISCH(K),
     1HEIGH(K)
  170 DO 180 K=IK,JK
      BCO(K)=ECO(K)
      BWIDT(K)=EWIDT(K)
      BAREA(K)=EAREA(K)
      BEDIS(K)=ENDIS(K)
      BEVEL(K)=ENVEL(K)
      BECEL(K)=ENCEL(K)
      BEDEP(K)=ENDEP(K)
  180 BETIM(K)=ENTIM(K)
      IP=2
      IF(DIST(IK)-XFIN/5280.) 185,183,183
  183 IK=IK+1
  185 IF(BEDIS(JK)-XFIN) 190,190,206
  190 GO TO (195,200),INDEX
  195 JK=JK+1
  200 J=J+1
206   IF(TIME(IK)-(TFIN/3600.)) 210,210,220
210   GO TO 36
220   CALL EXIT
      END
      SUBROUTINE CHAR2
C
C   COMPUTE THE PRESENT VALUES OF DEPTH AND VELOCITY FOR THE
C     CHANNEL UPSTREAM END
      COMMON EXCEL(830),EXTIM(830),BETIM(830),BEDIS(830),BDG(800),
     1    TIME(830),BECEL(830),ENTIM(830),ENDIS(830),ENVEL(830),
     2    DIST(830),BAREA(830),EAREA(830),BWIDT(800),EWIDT(830),
     3    DISCH(830),HEIGH(830),BEDEP(800),ENDEP(830),BEVEL(800),
     4    BCO(800),ECO(800),Z(800),STAGE(800),ENCEL(800),EDG(800)
      COMMON J,TDIPH,JN,MJ,DT,V,LL,O1,G,GG,S,K,IK,NOTE,INDEX,ML,
     1    IP,SQ,M,JK,XFIN,XZERO,TZERO,DEPI,DEPN,VI,VN
      EJ=J
      EXTIM(J)=TZERO+(EJ-1.)*TDIPH
36    JN=JK-1
      MJ=JK-2
   38 ENTIM(IK)=BETIM(IK)+DT
```

```
      V=BEVEL(IK)+BACEL(IK)
   42 ENDIS(IK)=BEDIS(IK)+V*(ENTIM(IK)-BETIM(IK))
      IF(ABS(ENDIS(IK)-XFIN)-1500.) 43,43,41
   41 IF(ENDIS(IK)-XFIN) 55,43,43
   43    CALL CHAR3
         GO TO 60
   55 EDG(IK)=(ENDIS(IK)-XZERO)/(XFIN-XZERO)
      ENDEP(IK)=DEPI+(DEPN-DEPI)*EDG(IK)
      D1=ENDEP(IK)
      ECO(IK)=CI(D1)+(CN(D1)-CI(D1))*EDG(IK)
      FAREA(IK)=A(D1)
      FWIDT(IK)=DAY(D1)
      ENCEL(IK)=SQRT(G*FAREA(IK)/FWIDT(IK))
      ENVEL(IK)=VI+(VN-VI)*EDG(IK)
      TIME(IK)=ENTIM(IK)/3600.
      DIST(IK)=ENDIS(IK)/5280.
C     CALCULATING  MIDDLE  POINTS
60    CONTINUE
      RETURN
      END
      SUBROUTINE  CHAR3
      COMMON EXCEL(830),EXTIM(830),BETIM(830),BEDIS(830),BDG(800),
     1    TIME(830),BACEL(830),ENTIM(830),ENDIS(830),ENVEL(830),
     2    DIST(830),BAREA(830),FAREA(830),BWIDT(800),FWIDT(830),
     3    DISCH(830),HEIGH(830),BEDEP(800),ENDEP(830),BEVEL(800),
     4    BCO(800),ECO(800),Z(800),STAGE(800),ENCEL(800),EDG(800)
      COMMON J,TDIPH,JN,MJ,DT,V,LL,D1,G,GG,S,K,IK,NOTE,INDEX,ML,
     1    IP,SQ,M,JK,XFIN,XZERO,TZERO,DEPI,DEPN,VI,VN
    2 FORMAT (1H I3,7E15.4)
C
C COMPUTING THE END POINT VALUES
   43 ENDIS(IK)=XFIN
      ENTIM(IK)=BETIM(IK)+(ENDIS(IK)-BEDIS(IK))/V
      V=BEVEL(IK)
      LL=1
      D1=ENDEP(IK)
   44 BO=B(D1)
      EDG(IK)=(ENDIS(IK)-XZERO)/(XFIN-XZERO)
      ECO(IK)=CN(D1)
      FAREA(IK)=A(D1)
      FWIDT(IK)=DAY(D1)
      U=BO/FAREA(IK)
      DADY=DAY(D1)
      DBDY=DBY(D1)
      DQDY=DQ(D1)
      DDAY=DDA(D1)
      V1=SQRT(FAREA(IK)+BAREA(IK))
      V2=D1-BEDEP(IK)
      V3=SQRT(FWIDT(IK)+BWIDT(IK))
      V4=(B(D1)/FAREA(IK))**(4./3.)
      V5=(B(BEDEP(IK))/BAREA(IK))**(4./3.)
      V6=ECO(IK)*U
      V7=ENTIM(IK)-BETIM(IK)
      V8=BCO(IK)*V
      V9=FWIDT(IK)/(V1*V1)
      V10=DDAY/(V3*V3)
```

```
      V11=SQ*(U+V)*EWIDT(IK)/(V1*V1*V1*V1)
      V12=(4./3.)*(DUDY/B(D1)-EWIDT(IK)/EAREA(IK))
      V13=.5*GG*V2/V1
      DUDY=(EAREA(IK)*DUDY-EQ*DADY)/(EAREA(IK)*EAREA(IK))
      SF1=.5*G/2.2082*(V6*V6*V4+V8*V8*V5)
      SF11=.5*G/2.2082*V6*V6*V4
      R1=U-V+2.*V3*V13-(SQ*2.*GG/(V1*V3)-SQ*(U+V)/(V1*V1)+G*S-SF1)
     1     *V7
      DR1DY=DUDY+GG*V3/V1+V13*(DDAY/V3-V3*V9)+(SQ*GG/(V3*V1)*
     1    (V9+V10)-V11+SF11*V12)*V7
      ENDEP(IK)=D1-R1/DR1DY
      ENVEL(IK)=U
      D1=ENDEP(IK)
      EAREA(IK)=A(D1)
      EWIDT(IK)=DAY(D1)
      IF((ABS(D1-ENDEP(IK)))-0.001)48,45,45
   45 IF(LL-3) 46,47,47
   46 LL=LL+1
      GO TO 44
   47 WRITE(3,8) LL,ENDEP(IK)
   48 DIST(IK)=ENDIS(IK)/5280.0
      TIME(IK)=ENTIM(IK)/3600.
      ENCEL(IK)=SQRT(G*EAREA(IK)/EWIDT(IK))
      RETURN
      END
      SUBROUTINE  CHAR4
C
C   CALCULATING THE INTERIOR POINTS ALONG THE CHARACTERICS
      COMMON EXCEL(830),EXTIM(830),BETIM(830),BEDIS(830),BDG(800),
     1    TIME(830),BECEL(830),ENTIM(830),ENDIS(830),ENVEL(830),
     2    DIST(830),BAREA(830),EAREA(830),BWIDT(800),EWIDT(830),
     3    DISCH(830),BETCH(830),BEDEP(800),ENDEP(830),BEVEL(800),
     4    BQQ(800),EQQ(800),Z(800),STAGE(800),ENCEL(800),EDG(800)
      COMMON J,TDIPH,JN,MJ,DT,V,LL,D1,G,GG,S,K,IK,NOTE,INDEX,ML,
     1    TP,SQ,M,JK,XEIN,XZERO,TZERO,DEPI,DEPN,VI,VN
    8 FORMAT (1H I3,7E15.6)
      AS=ENVEL(K)-ENCEL(K)
      BS=BEVEL(K+1)+BECEL(K+1)
      D=AS*ENTIM(K)-ENDIS(K)
      E=BS*BETIM(K+1)-BEDIS(K+1)
      ENTIM(K+1)=(D-E)/(AS-BS)
      ENDIS(K+1)=ENDIS(K)+AS*(ENTIM(K+1)-ENTIM(K))
      EDG(K+1)=(ENDIS(K+1)-XZERO)/(XEIN-XZERO)
      DD=0.5*(BEDEP(K+1)+ENDEP(K))
      VV=0.5*(BEVEL(K+1)+ENVEL(K))
      LL=1
      AA=BEVEL(K+1)
      BA=ENVEL(K)
      C1=-G*S*(ENTIM(K+1)-BETIM(K+1))
      C2=-G*S*(ENTIM(K+1)-ENTIM(K))
      C3=ENTIM(K+1)-BETIM(K+1)
      C4=ENTIM(K+1)-ENTIM(K)
      C5=G/2.2082
      D2=BEDEP(K+1)
      D3=ENDEP(K)
      A2=A(D2)
```

```
      A3=A(D3)
      B2=B(D2)
      B3=B(D3)
      BA2=DAY(D2)
      BA3=DAY(D3)
      G1=C5*BCO(K+1)*BCO(K+1)*C3
      G2=C5*ECO(K)*ECO(K)*C4
      G3=(B2/A2)**(4./3.)
      G4=(B3/A3)**(4./3.)
   72 CONTINUE
      A1=A(DD)
      B1=B(DD)
      BA1=DAY(DD)
      ECO(K+1)=CI(DD)+(CN(DD)-CI(DD))*EDG(K+1)
      UA1=SQRT(A1+A2)
      VA1=SQRT(A1+A3)
      UA2=SQRT(B1+B2)
      VA2=SQRT(B1+B3)
      UA3=DD-D2
      VA3=DD-D2
      UA4=A1+A2
      VA4=A1+A3
      UA5=B1+B2
      VA5=B1+B3
      UA=GG*UA3*UA2/UA1
      VA=GG*VA3*VA2/VA1
      G5=C5*ECO(K+1)*ECO(K+1)
      G6=(B1/A1)**(4./3.)
      UA6=UA4*UA4
      VA6=VA4*VA4
      TA=.5*(G5*G6*VV*ABS(VV)*C4+G2*BA*ABS(BA)*C4)
      SA=.5*(G5*G6*VV*ABS(VV)*C3+G1*AA*ABS(AA)*G3)
      Q1=SQ*C3*(2.*GG/(UA1*UA2)-(VV+AA)/UA4)
      Q2=SQ*C4*(2.*GG/(VA1*VA2)-(VV+BA)/VA4)
      R1=VV-AA+UA+G1+SA-Q1
      R2=VV-BA-VA+G2+TA-Q2
      DADY=R1
      DDAY=DDA(DD)
      DBDY=DBY(DD)
      DUA1=DDAY/UA2-UA2*DADY/UA4
      DVA1=DDAY/VA2-VA2*DADY/VA4
      DUA=GG*UA2/UA1+.5*GG*UA3/UA1*DUA1
      DVA=GG*VA2/VA1+.5*GG*VA3/VA1*DVA1
      DSA=G5*G6*VV*ABS(VV)*(4./3.)*(DBDY/B1-DADY/A1)
      DQ1U=SQ*C3/UA4
      DQ2U=SQ*C4/VA4
      DQ1Y=SQ*C3*(GG/(UA1*UA2)*(DADY/UA4+DDAY/UA5)-(VV+AA)*DADY/UA6)
      DQ2Y=SQ*C4*(GG/(VA1*VA2)*(DADY/VA4+DDAY/VA5)-(VV+BA)*DADY/VA6)
      DR1DY=DUA+DQ1Y+DSA*C3
      DR2DY=-DVA+DQ2Y+DSA*C4
      DR1DU=1.+G5*G6*VV*C3+DQ1U
      DR2DU=1.+G5*G6*VV*C4+DQ2U
      DDUY=DR1DU*DR2DY-DR1DY*DR2DU
      DU=(R1*DR2DY-R2*DR1DY)/DDUY
      DY=(R2*DR1DU-R1*DR2DU)/DDUY
      ENVEL(K+1)=VV-DU
```

```
      ENDEP(K+1)=DD-DY
      LL=LL+1
      IF(LL-10) 75,80,80
   75 IF(ABS(VV-ENVEL(K+1))-0.005) 76,76,79
   76 IF(ABS(DD-ENDEP(K+1))-0.005) 81,79,79
   79 VV=ENVEL(K+1)
      DD=ENDEP(K+1)
      GO TO 72
   80 KK=K+1
      WRITE(3,8) KK,ENVEL(K+1),ENDEP(K+1)
   81 ENDIS(K+1)=ENDIS(K)+AS*(ENTIM(K+1)-ENTIM(K))
      D4=ENDEP(K+1)
      FAREA(K+1)=A(D4)
      FWIDT(K+1)=DAY(D4)
      ENCEL(K+1)=SQRT(G*FAREA(K+1)/FWIDT(K+1))
      DIST(K+1)=ENDIS(K+1)/5280.
      TIME(K+1)=ENTIM(K+1)/3600.
      RETURN
      END
      SUBROUTINE  CHARS
C
C    CALCULATING POINTS ALONG T-AXIS (UPSTREAM BOUNDARY)
      COMMON EXCEL(830),EXTIM(830),BETIM(830),BEDIS(830),BDG(800),
     1     TIME(830),BECEL(830),ENTIM(830),ENDIS(830),ENVEL(830),
     2     DIST(830),BAREA(830),FAREA(830),BWIDT(800),EWIDT(830),
     3     DISCH(830),HEIGH(830),BEDEP(800),ENDEP(830),BEVEL(800),
     4     BQU(800),EQQ(830),Z(800),STAGE(800),ENCEL(800),EDG(800)
      COMMON J,IDIPH,JM,MJ,DT,V,LL,D1,G,GG,S,K,IK,NOTE,INDEX,ML,
     1     IP,SO,M,JK,XFIN,XZERO,TZERO,DEPI,DEPN,VI,VN
    8 FORMAT (1H I3,7E15.6)
      EDG(JK)=(ENDIS(JK)-XZERO)/(XFIN-XZERO)
      V=ENVEL(JN)
      D1=.5*(ENDEP(JK)+ENDEP(JN))
      U=.5*(ENVEL(JN)+BEVEL(JN))+.5*ABS(ENVEL(JN)-BEVEL(JN))
      EQQ(JK)=CI(D1)+(CM(D1)-CI(D1))*EDG(JK)
      FAREA(JK)=A(D1)
      FWIDT(JK)=DAY(D1)
      LL=1
  100 CONTINUE
      V1=SQRT(FAREA(JK)+FAREA(JM))
      V2=D1-ENDEP(JN)
      V3=SQRT(FWIDT(JK)+FWIDT(JN))
      V4=(P(D1)/FAREA(JK))**(4./3.)
      V5=(P(ENDEP(JN))/FAREA(JN))**(4./3.)
      V6=EQQ(JK)*U
      V7=ENTIM(JK)-ENTIM(JN)
      V8=EQQ(JN)*V
      V13=.5*GG*V2/V1
      SF1=.5*G/2.2082*(V5*V6*V4+V8*V3*V5)
      R2=U-V-2.*V3*V13-(SO*2.*GG/(V1*V3)-SO*(U+V)/(V1*V1)+G*S-SF1)*V7
      DR2DU=1.+V7*(SO/(V1*V1)+G*V6*V4*EQQ(JK)/2.2082)
      ENVEL(JK)=U-R2/DR2DU
      AS=.5*(ENVEL(JN)-ENCEL(JN))+.5*(ENVEL(JK)-ENCEL(JK))
      ENTIM(JK)=ENTIM(JN)+(ENDIS(JK)-ENDIS(JN))/AS
      ENDEP(JK)=STAGE(ML)+(ENTIM(JK)-EXTIM(ML))*(STAGE(ML)-STAGE(ML-1))/
     1TDIPH
```

```
      D1=ENDEP(JK)
      EDD(JK)=CI(D1)+(CN(D1)-CI(D1))*EDG(JK)
      EAREA(JK)=A(D1)
      EWIDT(JK)=DAY(D1)
      DIST(JK)=ENDIS(JK)/5280.
      TIME(JK)=ENTIM(JK)/3600.
      LL=LL+1
      IF(ABS(U-ENVEL(JK))-0.01) 121,121,118
  118 IF(LL-10) 119,120,120
  119 U=ENVEL(JK)
      GO TO 100
  120 WRITE(3,8) JK,ENVEL(JK)
  121 DIST(JK)=ENDIS(JK)/5280.
      ENDEP(JK)=STAGE(ML)+(ENTIM(JK)-EXTIM(ML))*(STAGE(ML)-STAGE(ML-1))/
     1TDIFH
      RETURN
      END
      FUNCTION A(Y)
      A=2.20489*Y*Y*Y-8.02071*Y*Y+190.635*Y+228.779
      RETURN
      END
      FUNCTION B(Y)
      B=(((-.0345819*Y+1.74116)*Y-21.1231)*Y+131.725)*Y+28.3223
      RETURN
      END
      FUNCTION Q(Y)
      Q=-.324372*Y*Y*Y*Y+21.5435*Y*Y*Y-413.972*Y*Y+3668.52*Y-9335.86
      RETURN
      END
      FUNCTION CI(Y)
      CI=(((.281877E-05*Y-.24522E-04)*Y+.00076281)*Y-.0102767)*Y+
     10.0614686)*Y-0.0876877
      RETURN
      END
      FUNCTION CN(Y)
      CN=(((-.164875E-5*Y+.968542E-4)*Y-.00181375)*Y+.0135891)*Y+.005855
      RETURN
      END
      FUNCTION DAY(Y)
      DAY=6.61467*Y*Y-16.0414*Y+190.635
      RETURN
      END
      FUNCTION DDY(Y)
      DDY=((-0.1383276*Y+5.22343)*Y-42.2462)*Y+131.725
      RETURN
      END
      FUNCTION DQ(Y)
      DQ=-1.297483*Y*Y*Y+14.5305*Y*Y-829.944*Y+3668.52
      RETURN
      END
      FUNCTION DDA(Y)
      DDA=13.22934*Y-16.0414
      RETURN
      END

  120    1    2    2    2    1    1
```

FLOOD ROUTING IN THE NATURAL CHANNEL

BY THE IMPLICIT METHOD

(Composite Section)

```
C                    STREAMFLOW  ROUTING
C     THIS  1945  FLOOD  AT  NEUSE  RIVER START SEPT    3 O'CLOCK
C                      IMPILIT  METHOD
      COMMON    Z(150),DIST(150),VIN(150),VIN(150),EV(99),EY(99),X(150),
     1C(31),DV(150),DY(150),A(31,31),C1(31),STAGE(150),EH(99),HIN(150),
     2DG(150),DISCH(150),T(150),VT(100),YT(100),Q(130),EQ(130),EA(130)
      COMMON   TEIN,TZERO,G,DTIME,XZERO,XEIN,ELEVO,ELEVN,DN,DISHI,DISHN,
     1DELX,DELT,MN,N,L,SO,SO,MI,MA,DTDX,DXDT,DX,DT,TIME,I,DR1,DR2,U,V,R,
     2S,DR1DU,DR1DV,DR1DS,DR1DT,DR2DU,DR2DV,DR2DS,DR2DT
    3 FORMAT(8E10.3)
    4 FORMAT(///35X,'STREAMFLOW ROUTING BY THE IMPILIT METHOD'///)
5     FORMAT(16I5)
   11 FORMAT(55X,'COEFFICIENTS OF THE MATRIX')
   12 FORMAT(/// 11(2X,E9.4))
   13 FORMAT(///1X,11(2X,E9.4))
   21 FORMAT(///55X,'VALUES OF VARIABLES')
   23 FORMAT(///15X,'TIME',15X,'DISTANCE',15X,'DEPTH',13X,'VELOCITY')
   25 FORMAT(1H,4(5X,E15.8))
C     WETTED PERIMETER IS EQUAL TO THE TOP WIDTH
C     INITIAL   DATA
      READ(1,3)TEIN,TZERO,G,DTIME,XZERO,XEIN,ELEVO,ELEVN,DN,DISHI,DISHN
     1 ,DELX,DELT
      GG=SQRT(G)
      READ(1,3)(STAGE(K),K=1,MA)
      READ(1,5)MN,N,L
      WRITE(3,3)(STAGE(K),K=1,MA)
      WRITE(3,3)TEIN,TZERO,G,DTIME,XZERO,XEIN,ELEVO,ELEVN,DN,DISHI,DISHN
     1 ,DELX,DELT
      WRITE(3,4)
      WRITE(3,5)MN,N,L
      SO=(ELEVO-ELEVN)/(5280.*XEIN)
      DI=STAGE(1)
      SO=(DISHN-DISHI)/(XEIN*5280.)
      VI=DISHI/AR(DI)
      VN=DISHN/AR(DN)
C     DELX IS GIVEN IN MILES AND DELT IS GIVEN IS HOURS
      NL=2.*N-1
      NN=N-1
      Z(1)=ELEVO
      T(1)=TZERO
      DIST(1)=0.0
      DO 30 I=1,NN
      DIST(I+1)=DIST(I)+DELX
   30 Z(I+1)=ELEVO-(DIST(I+1)-XZERO)*SO*5280.0
      DT=DELT*3600.0
      DX=DELX*5280.
      DXDT=DX/DT
      DTDX=DT/DX
      TIME=TZERO
C     INITIAL VALUES
      DO 50 I=1,N
      DG(I)=(DIST(I)-XZERO)/(XEIN-XZERO)
      VIN(I)=VI+(VN-VI)*DG(I)
      YIN(I)=DI+(DN-DI)*DG(I)
      EV(I)=VIN(I)
```

```
      EY(I)=YIN(I)
      HIN(I)=YIN(I)+Z(I)
      EH(I)=HIN(I)
      DISCH(I)=DISHI+(DISHN-DISHI)*CG(I)
      Q(I)=S0
   50 CONTINUE
C     COMPUTATIONS AT UPSTREAM BOUNDARY
      MM=MN-1
      DO 60 K=1,MM
   60 T(K+1)=T(K)+DTIME
      TIME=TIME+DELT
65    CALL IMPL3
      CALL IMPL4
      CALL IMPL5
      DO 180 I=1,NL
  180 C1(I)=C(I)
      CALL MSIMQ(A,C1,NL,KS)
      DO 190 I=1,NL
  190 X(I)=C1(I)
      DO 200 I=1,N
      VT(I)=EV(I)
  200 YT(I)=EY(I)
      K=NL-1
      DO 210 J=1,K,2
      I=(J+1)/2
      EV(I)=X(J)
  210 EY(I+1)=X(J+1)
      EV(N)=X(NL)
      DO 240 I=1,N
      ERROR=EV(I)-VT(I)
      IF(ABS(ERROR)-0.02) 240,240,65
  240 CONTINUE
      WRITE(3,21)
      WRITE(3,23)
      DO 250 I=1,N
  250 WRITE(3,25) TIME,DIST(I),EY(I),EV(I)
      DO 260 I=1,N
      DY(I)=EY(I)-YIN(I)
  260 DV(I)=EV(I)-VIN(I)
      DO 270 I=1,N
      YIN(I)=EY(I)
  270 VIN(I)=EV(I)
      TIME=TIME+DELT
      IF(TIME-TFIN) 280,300,300
280   DO 285 I=1,N
      EY(I)=EY(I)+DY(I)*.5
285   EV(I)=EV(I)+DV(I)*.5
      GO TO 65
300   CALL EXIT
      END
      SUBROUTINE IMPL3
      COMMON    Z(150),DIST(150),YIN(150),VIN(150),EV(99),EY(99),X(150),
     1C(31),DV(150),DY(150),A(31,31),C1(31),STAGE(150),EH(99),HIN(150),
     2CG(150),DISCH(150),T(150),VT(100),YT(100),Q(130),EQ(130),EA(130)
      COMMON   TFIN,TZERO,G,DTIME,XZERO,XFIN,ELEVO,ELEVN,DN,DISHI,DISHN,
     1DELX,DELT,MN,N,L,SC,SQ,NL,MM,DTDX,DXDT,DX,DT,TIME,I,DR1,DR2,U,V,R,
```

```
     2S,DR1DU,DR1DV,DR1DS,DR1DT,DR2DU,DR2DV,DR2DS,DR2DT
  65 DO 70 I=1,NL
     X(I)=0.0
     DO 70 J=1,NL
  70 A(I,J)=0.0
     IF(TIME-T(L)) 74,71,72
  71 SUREL=STAGE(L)
     GO TO 80
  72 L=L+1
  74 SUREL=STAGE(L-1)+(TIME-T(L-1))*(STAGE(L)-STAGE(L-1))/DTIME
     EY(1)=SUREL
  80 EH(1)=SUREL+ELEVO
     I=1
     CALL IMPL7
     A(1,1)=DR1DU
     A(1,2)=DR1DT
     A(1,3)=DR1DV
     A(2,1)=DR2DU
     A(2,2)=DR2DT
     A(2,3)=DR2DV
     C(1)=DR1DU*U+DR1DV*V+DR1DT*R-DR1
     C(2)=DR2DU*U+DR2DV*V+DR2DT*R-DR2
     RETURN
     END
     SUBROUTINE IMPL4
     COMMON    Z(150),DIST(150),YIN(150),VIN(150),EV(99),EY(99),X(150),
    1C(31),DV(150),DY(150),A(31,31),C1(31),STAGE(150),EH(99),HIN(150),
    2DG(150),DISCH(150),T(150),VT(100),YT(100),Q(130),EQ(130),EA(130)
     COMMON   TEIN,TZERO,G,DTIME,XZERO,XEIN,ELEVO,ELEVN,DN,DISHI,DISHN,
    1DELX,DELT,MN,N,I,SO,SQ,NL,NN,DTDX,DXDT,DX,DT,TIME,I,DR1,DR2,U,V,R,
    2S,DR1DU,DR1DV,DR1DS,DR1DT,DR2DU,DR2DV,DR2DS,DR2DT
C    COMPUTATIONS  AT INTERIOR POINT
 100 DO 150 I=2,NN
     CALL IMPL7
     J=2.*I-2.
     M=2.*I-1
     A(M,J)=DR1DS
     A(M,J+1)=DR1DU
     A(M,J+2)=DR1DT
     A(M,J+3)=DR1DV
     A(M+1,J)=DR2DS
     A(M+1,J+1)=DR2DU
     A(M+1,J+2)=DR2DT
     A(M+1,J+3)=DR2DV
     C(M)=DR1DU*U+DR1DV*V+DR1DT*R-DR1+DR1DS*S
     C(M+1)=DR2DU*U+DR2DV*V+DR2DT*R-DR2+DR2DS*S
 150 CONTINUE
     RETURN
     END
     SUBROUTINE IMPL5
C    COMPUTATION  AT  DOWNSTREAM  BOUNDARY
     COMMON    Z(150),DIST(150),YIN(150),VIN(150),EV(99),EY(99),X(150),
    1C(31),DV(150),DY(150),A(31,31),C1(31),STAGE(150),EH(99),HIN(150),
    2DG(150),DISCH(150),T(150),VT(100),YT(100),Q(130),EQ(130),EA(130)
     COMMON   TEIN,TZERO,G,DTIME,XZERO,XEIN,ELEVO,ELEVN,DN,DISHI,DISHN,
    1DELX,DELT,MN,N,I,SO,SQ,NL,NN,DTDX,DXDT,DX,DT,TIME,I,DR1,DR2,U,V,R,
```

```
2S,DR1DU,DR1DV,DR1DS,DR1DT,DR2DU,DR2DV,DR2DS,DR2DT
 P=EV(N)
 R=EY(N)
 DADY=DAY(R)
 D1=0.498982E-16
 D2=0.443349E-11
 D3=0.146063E-06
 D4=0.247673E-02
 D5=1.75506
 V1=P*P
 A1=AR(P)
 A2=A1*A1
 W=P*A1
 DR1=P+D1*W**4-D2*W**3+D3*W*W-D4*W-D5
 DR1DV=A1*(D1*A2*A1*4.*V1*P-3.*D2*V1*A2+2.*D3*A1*P-D4)
 DR1DR=1.+DADY*P*(4.*D1*V1*P*A1*A2-3.*D2*V1*A2+2.*D3*P*A1-D4)
 A(NL,NL-1)=DR1DR
 A(NL,NL)=DR1DV
 C(NL)=DR1DR*R+DR1DV*P-DR1
 RETURN
 END
 SUBROUTINE IMPLZ
 COMMON     Z(150),DIST(150),YIN(150),VIN(150),EV(99),EY(99),X(150),
1C(31),DV(150),DY(150),A(31,31),CI(31),STAGE(150),EH(99),HIN(150),
2DG(150),DISCH(150),T(150),VT(100),YT(100),Q(130),EQ(130),EA(130)
 COMMON   TEIN,TZERO,G,CTIME,XZERO,XEIN,ELEVO,ELEVN,DN,DISHI,DISHN,
1DELX,DELT,MN,N,L,SO,SQ,NL,NN,DTDX,DXDT,DX,DT,TIME,I,DR1,DR2,U,V,R,
2S,DR1DU,DR1DV,DR1DS,DR1DT,DR2DU,DR2DV,DR2DS,DR2DT
 U=EV(I)
 V=EV(I+1)
 S=EY(I)
 R=EY(I+1)
 SB=YIN(I)
 RB=YIN(I+1)
 UB=VIN(I)
 VB=VIN(I+1)
 A1=AR(S)
 A2=AR(R)
 B1=BR(S)
 B2=BR(R)
 AB1=AR(SB)
 AB2=AR(RB)
 BB1=BR(SB)
 BB2=BR(RB)
 CB1=CI(SB)+(CN(SB)-CI(SB))*DG(I)
 CB2=CI(RB)+(CN(RB)-CI(RB))*DG(I+1)
 CI1=CI(S)
 CN1=CN(S)
 CI2=CI(R)
 CN2=CN(R)
 C01=CI1+(CN1-CI1)*DG(I)
 C02=CI2+(CN2-CI2)*DG(I+1)
 Q1=Q(I)
 Q2=Q(I+1)
 DA1DY=DAY(S)
 DA2DY=DAY(R)
```

```
      DB1DY=DBY(S)
      DB2DY=DBY(R)
      DB3DY=DDAY(S)
      DB4DY=DDAY(R)
      DA3DY=DAY(SB)
      DA4DY=DAY(RB)
      V1=V+VB-U-UB
      V2=V+U-VB-UB
      D1=R-S+RB-SB
      D2=R+S-SB-RB
      VD=V+U+VB+UB
      E1=DA1DY
      E2=DA2DY
      EB1=DA3DY
      EB2=DA4DY
      QD=Q1*(U/A1+V/A2+UB/AB1+VB/AB2)
      BD=1./E1+1./E2+1./EB1+1./EB2
      AD=A1/E1+A2/E2+AB1/EB1+AB2/EB2
      SE1=C01*C01*U*ABS(U)*(B1/A1)**(4./3.)/2.2082
      SE2=C02*C02*V*ABS(V)*(B2/A2)**(4./3.)/2.2082
      SE3=CB1*CB1*UB*ABS(UB)*(BE1/AB1)**(4./3.)/2.2082
      SE4=CB2*CB2*VB*ABS(VB)*(BB2/AB2)**(4./3.)/2.2082
      DSE2=C02*C02*V*(B2/A2)**(4./3.)/2.2082
      DSE1=C01*C01*U*(B1/A1)**(4./3.)/2.2082
      SE=SE1+SE2+SE3+SE4
      DB4=DB1DY/B1
      DB3=DB3DY/(E1*E1)
      DB2=DB4DY/(E2*E2)
      DB1=DB2DY/B2
      DA4=DA1DY/A1
      DA3=DA1DY/(A1*A1)
      DA2=DA2DY/(A2*A2)
      DA1=DA2DY/A2
      DR1=.25*DTDX*VD*D1+D2+DTDX*.25*V1*AD-.5*Q1*DT*BD
      DR2=DXDT*V2/G+.25*VD*V1/G+.5*DX*QD/G+D1-DX*2.*SD+.5*DX*SE
      DR1DS=1.-.25*DTDX*VD+DTDX*V1*(DA1DY/E1-A1*DB3)*.25+.5*DT*DB3*Q1
      DR1DT=.25*DTDX*VD+1.+DTDX*V1*(DA2DY/E2-A2*DB2)*.25+.5*DT*Q1*DB2
      DR1DV=.25*DTDX*D1+DTDX*AD*.25
      DR1DU=.25*DTDX*D1-DTDX*AD*.25
      DR2DU=(DXDT-.5*(U+UB)+.5*DX*Q1/A1)/G+DX*DSE1
      DR2DV=(DXDT+.5*(V+VB)+.5*DX*Q1/A2)/G+DX*DSE2
      DR2DT=1.-.5*DX*Q1*V*DA2/G+DX*(4./3.)*SE2*(DB1-DA1)*.5
      DR2DS=-1.-.5*DX*Q1*U*DA3/G+DX*(4./3.)*SE1*(DB4-DA4)*.5
      RETURN
      END
      SUBROUTINE MSIMC (A,P,N,KS)
      DIMENSION A(1),P(1)
C
C               FORWARD  SOLUTION
      TOL=0.0
      KS=0
      JJ=-N
      DO 65 J=1,N
      IY=J+1
      JJ=JJ+N+1
      BIGA=0.
```

```
      IT=JJ-J
      L1=J+4
      IF(L1-N) 10,10,12
   12 L1=N
   10 DO 30 I=J,L1
C
C           SEARCH FOR MAXIMUM COEFFICIENT IN COLUMN
C
      IJ=IT+I
      IF(ABS(BIGA)-ABS(A(IJ))) 20,30,30
   20 BIGA=A(IJ)
      IMAX=I
   30 CONTINUE
C
C         TEST FOR PIVOT LESS THAN TOLENANCE(SINGULAR MATRIX)
C
   32 IF(ABS(BIGA)-TOL) 35,35,40
   35 KS=1
      RETURN
C
C           INTERCHANGE ROWS IF NECESSARY
C
   40 I1=J+N*(J-2)
      IT=IMAX-J
      L2=J+4
      IF(L2-N) 42,42,44
   44 L2=N
   42 DO 50 K=J,L2
      I1=I1+N
      I2=I1+IT
      SAVE=A(I1)
      A(I1)=A(I2)
      A(I2)=SAVE
C
C           DIVIDE EQUATION BY LEADING COEFFICIENT
C
   50 A(I1)=A(I1)/BIGA
   52 SAVE=B(IMAX)
      B(IMAX)=B(J)
      B(J)=SAVE/BIGA
C
C           ELIMINATE NEXT VARIABLE
C
      IF(J-N) 55,70,55
   55 IQS=N*(J-1)
      L3=JY+4
      IF(L3-N) 57,57,59
   59 L3=N
   57 DO 65 IX=JY,L3
      IXJ=IQS+IX
      IT=J-IX
      DO 60 JX=JY,L3
      IXJX=N*(JX-1)+IX
      JJX=IXJX+IT
   60 A(IXJX)=A(IXJX)-(A(IXJ)*A(JJX))
   65 B(IX)=B(IX)-(B(J)*A(IXJ))
```

```
C
C          BACK  SOLUTION
C
   70 NY=N-1
      IT=N*N
      DO 80 J=1,NY
      IA=IT-J
      IB=N-J
      IC=N
      DO 80 K=1,J
      B(IB)=B(IB)-A(IA)*B(IC)
      IA=IA-N
   80 IC=IC-1
      RETURN
      END
      FUNCTION AR(Y)
      AR=2.20489*Y*Y*Y-8.02071*Y*Y+190.635*Y+228.779
      RETURN
      END
      FUNCTION BR(Y)
      BR=(((-.0345819*Y+1.74116)*Y-21.1231)*Y+131.725)*Y+28.3223
      RETURN
      END
      FUNCTION BO(Y)
      BO=-.324372*Y*Y*Y*Y+21.5435*Y*Y*Y-413.972*Y*Y+3668.52*Y-9335.86
      RETURN
      END
      FUNCTION CI(Y)
      CI=((((.2818775E-06*Y-.24522E-04)*Y+.00076281?)*Y-.0102767)*Y+
     10.0614686)*Y-0.0876877
      RETURN
      END
      FUNCTION CN(Y)
      CN=(((-.1661?7E-5*Y+.96?542E-4)*Y-.0018?375)*Y+.0135891)*Y+.005855
      RETURN
      END
      FUNCTION DAY(Y)
      DAY=6.61467*Y*Y-16.04142*Y+190.635
      RETURN
      END
      FUNCTION DBY(Y)
      DBY=((-0.1383276*Y+5.22343)*Y-42.2462)*Y+131.725
      RETURN
      END
      FUNCTION DDAY(Y)
      DDAY=13.22934*Y-16.04142
      RETURN
      END
      FUNCTION DOY(Y)
      DOY=-1.297488*Y*Y*Y+64.6305*Y*Y-829.944*Y+3668.52
      RETURN
      END
```

FLOOD ROUTING IN THE NATURAL CHANNEL BY THE IMPLICIT METHOD

(Discrete Section)

```
C                        IMPICIT   METHOD
C      THE 1945  FLOOD  AT  NEUSE  RIVER START SEPT 18, 3:00 A.M.
C      WETTED PERIMETER IS EQUAL TO THE TOP WIDTH
C    A(I,J)=MATRIX OF COEFFICIENTS
C    DELX =INCREMENT OF DISTANCE IN MILES
C    DELT =INCREMENT OF TIME  IN HOURS
C    G    =GRAVITATIONAL  CONSTANT
C    TFIN =FINAL TIME
C    MSIMQ=A SUBROUTINE FOR OBTAINING SOLUTION OF A SET OF
C      SIMULTANEOUS LINEAR EQUATICNS AX=C ,IT WAS OBTAINED BY
C      MODIFYING SIMQ WHICH IS A BUILT-IN PROGRAM IN IBM 360 SYSTEM
C     READ IN DATA AND FUNCTICNS THAT APPLY THIS PROGRAM
C     THIS PROGRAM SOLVES  THE SYSTEM CF NONLINEAR EQUATIONS
C      BY GENERALIZED NEWTCN RAPHSCN'S METHOD SIMULTANEOUSLY
C     IN THIS PROGRAM THE CROSS-SECTIONAL AREA,TOP-WIDTH,MANNING'S
C      ROUGHNESS COEFFICIENT, DCWNSTREAM END RATING CURVE AND THEIR
C      CORRESPONDING DERIVATIVES ARE ALL WRITTEN IN TERMS OF DEPTH
C      AS  FUNCTIONS OR AS ARITHMETIC  FUNCTIONS
C     DISHI =INITIAL DISCHARGE AT UPSTREAM END ,IN CUBFT PER SEC
C     DISHN =INITIAL DISCHARGE AT DCWNSTREAM END,IN CUB FT/SEC
C     XFIN  =FINAL  DISTANCE, IN MI
C     SQ    =LATERAL  INFLOW, IN  CUBIC FT PER SEC PER FT
C     DI    =INITIAL DEPTH AT  UPSTREAM END, IN FT
C     DN    =INITIAL DEPTH AT DCWNSTREAM END,IN FT
C     NL    =NUMBER OF EQUATICNS AND NUMBER OF UNKNOWNS
C     TIME  =TIME AT ANY INSTANT  IN SEC
C     T(K)  =VALUE  OF T ON T-AXIS AT EACH GIVEN POINT OF MN IN HR
C     C(K)  =VECTOR  OF RESIDUES (AFTER SUBTITUTED TRIAL VALUES)
C     Z(K)  =CHANNEL  BOTTCN HEIGHT IN FEET
C     DIST(K)=VALUE OF DISTANCE FCR EACH GRID POINT FROM X=0, IN MI
C     YIN(K)=DEPTH  AT   T=J  (T=TIME AT J POSITION)
C     VIN(K)=VELOCITY AT T=J
C     EV(K) =VELOCITY AT T=J+1
C     X(K)  =VECTOR FOR THE NEW TRIAL VALUES
C     SO    =CHANNEL  BOTTOM  SLOPE
C     STAGE(K)=INPUT HYDROGRAPH GIVEN IN  EVERY DTIME, IN FT
C     DG(K) =INTERPOLATION FUNCTICN IN TERMS OF DISTANCE
C     DISCH(K)=DISCHARGE  IN CUBIC FT PER SEC
C     EA(K) =AREA  AT EACH  GRID  POINT BASED ON DEPTH,IN SQUARE FT
C     Q(K)  =LATERAL INFLOW =SQ
C     EQ(K) =DISCHARGE AT EACH POINT BASED ON DEPTH,IN FT**3/SEC
C     MN    =NUMBER OF POINTS GIVEN  ON THE T-AXIS
C     N     =NUMBER OF POINTS GIVEN  CN THE X-AXIS
C     TZERO =INITIAL  TIME  IN HR
C     DTIME =TIME INTERVAL BETWEEN DISCRETE POINTS ON THE T-AXIS
C     XZERO =INITIAL  DISTANCE IN MI
C     ELEVO =CHANNEL BOTTOM HEIGHT AT UPSTREAM END, IN FT
C   L=AN INDEX FOR IDENTIFYING THE CURRENT VALUE OF TIME ON T-AXIS
C    AR=CROSSSECTIONAL AREA EQUATICNS. IT IS WRITTEN AS A DOUBLE
C      SUBSCRIBED FUNCTION. FIRST SUBSCRIBED VARIABLE STANDS FOR
C      DEPTH AND SECAND SUBSCRIBED INTEGER I STANDS FOR CALCULATING
C       GRID POINT
C    DBY=DERIVATIVE OF TOP-WIDTH BR W.R.T. DEPTH Y. I IS REFERS TO
C      THE CALCULATING GRID POINT. IT IS A DOUBLE SUBSCRIBED FUNCTION
C    DDAY=SECOND DERIVATIVE OF FUNCTICN AR  W.R.T. DEPTH Y.
```

```
C     IT IS WRITTEN AS A DOUBLE SUBSCRIBED FUNCTION AT GRID POINT I
C    DAY =FIRST DERIVATIVE OF FUNCTION AR W.R.T. DEPTH Y. IT IS
C    BR =TOP-WIDTH FUNCTION WHICH DEFINED AS SAME AS AR
C     DEFINED AS SAME AS AR
C
C
C          MAIN  PROGRAM
      COMMON    Z(150),DIST(150),YIN(150),VIN(150),EV(99),EY(99),
     1   C(31),DV(150),DY(150),A(31,31),C1(31),STAGE(150),EH(99),
     2   DG(150),DISCH(150),T(150),VT(160),YT(160),Q(160),EQ(160),
     3   X(150),EA(150)
      COMMON   G,DTIME,SO,DTDX,DXDT,DX,DT,TIME,DR1,DR2,S,R,V,U,
     1   DR1DU,DR1DV,DR1DS,DR1DT,DR2DU,DR2DV,DR2DS,DR2DT
      COMMON NN,N,L,NL,MN,I,MM
2     FORMAT(16I5)
    3 FORMAT(8F10.3)
    4 FORMAT(///35X,'STREAMFLOW ROUTING BY THE IMPICIT METHOD')
   11 FORMAT(55X,'COEFFICIENTS OF THE MATRIX')
   12 FORMAT(/// 11(2X,F9.4))
   21 FORMAT(///55X,'VALUES OF VARIABLES')
   23 FORMAT(///10X,'TIME',10X,'DISTANCE',10X,'DEPTH',10X,
     1'VELOCITY',8X,'DISCHARGE',8X,'AREA')
   25 FORMAT(1H,7(3X,E13.4))
C
C      INITIAL   DATA
C
      READ(1,2)MN,N,L
      IP=10
      READ(1,3)TFIN,TZERO,G,DTIME,XZERO,ELEVO,ELEVN,DISHI,DISHN,
     1    DELX,XFIN,DELT
      READ(1,3)(STAGE(K),K=1,MN)
      READ(1,3)(Z(K),K=1,N)
      READ(1,3)(YIN(K),K=1,N)
C     DELX IS GIVEN IN MILES AND DELT IS GIVEN IS HOURS
      WRITE(3,4)
      DN=6.85
      DI=STAGE(1)
      SO=(5900.-DISHI)/(XFIN*5280.)
      MM=MN-1
      NL=2.*N-1
      NN=N-1
      T(1)=TZERO
      DIST(1)=0.0
333   FORMAT (1H I2,5E14.6)
      ITRAY=1
      DO 30 I=1,NN
   30 DIST(I+1)=DIST(I)+DELX
      DT=DELT*3600.0
      DX=DELX*5280.
      DXDT=DX/DT
      DTDX=DT/DX
      TIME=TZERO
      DO 50 I=1,N
      DG(I)=(DIST(I)-XZERO)/(XFIN-XZERO)
      DISCH(I)=DISHI+(DISHN-DISHI)*DG(I)
      AA=AR(YIN(I),I)
```

```
      VIN(I)=DISCH(I)/AA
      EV(I)=VIN(I)
      EY(I)=YIN(I)
      Q(I)=SQ
   50 CONTINUE
      DO 60 K=1,MM
   60 T(K+1)=T(K)+DTIME
      TIME=TIME+DELT
65    CALL IMPL3
      CALL IMPL4
      CALL IMPL5
      DO 180 I=1,NL
  180 C1(I)=C(I)
      CALL MSIMQ(A,C1,NL,KS)
      DO 190 I=1,NL
  190 X(I)=C1(I)
      DO 200 I=1,N
      VT(I)=EV(I)
  200 YT(I)=EY(I)
      K=NL-1
      DO 210 J=1,K,2
      I=(J+1)/2
      EV(I)=X(J)
  210 EY(I+1)=X(J+1)
      EV(N)=X(NL)
      ITRAY=ITRAY+1
      IF(ITRAY-50)330,330,335
335   WRITE(3,333) ITRAY,EY(1),EY(N),EV(1),EV(N)
      CALL EXIT
330   CONTINUE
      DO 240 I=1,N
      ERROR=EV(I)-VT(I)
      IF(ABS(ERROR)-0.02) 240,240,65
  240 CONTINUE
      IF(IP-TIME) 245,245,255
  245 CONTINUE
      IP=IP+10
      WRITE(3,21)
      WRITE(3,23)
      DO 250 I=1,N
      EA(I)=AR(EY(I),I)
      EQ(I)=EA(I)*EV(I)
  250 WRITE(3,25) TIME,DIST(I),EY(I),EV(I),EQ(I),EA(I)
      ITRAY=1
  255 CONTINUE
      DO 260 I=1,N
      DY(I)=EY(I)-YIN(I)
  260 DV(I)=EV(I)-VIN(I)
      DO 270 I=1,N
      YIN(I)=EY(I)
  270 VIN(I)=EV(I)
  280 TIME=TIME+DELT
      DO 285 I=1,N
      EY(I)=EY(I)+DY(I)*.5
      EV(I)=EV(I)+DV(I)*.5
  285 CONTINUE
```

```
      IF(TIME-TFIN) 290,300,300
290   GO TO 65                                                          100
300   CALL EXIT
      END
      SUBROUTINE  IMPL3
      COMMON    Z(150),DIST(150),YIN(150),VIN(150),EV(99),EY(99),
     1   C(31),DV(150),DY(150),A(31,31),C1(31),STAGE(150),EH(99),
     2   DG(150),DISCH(150),T(150),VT(160),YT(160),Q(160),EQ(160),
     3   X(150),EA(150)
      COMMON   G,DTIME,SC,DTDX,DXDT,DX,DT,TIME,DR1,DR2,S,R,V,U,
     1   DR1DU,DR1DV,DR1DS,DR1DT,DR2DU,DR2DV,DR2DS,DR2DT
      COMMON NN,N,L,NL,MN,I,MM
C     COMPUTATIONS AT UPSTREAM BOUNDARY
   65 DO 70 I=1,NL
      X(I)=0.0
      DO 70 J=1,NL
   70 A(I,J)=0.0
      IF(TIME-T(L)) 74,71,72
   71 SUREL=STAGE(L)
      GO TO 80
   72 L=L+1
   74 SUREL=STAGE(L-1)+(TIME-T(L-1))*(STAGE(L)-STAGE(L-1))/DTIME
   80 EY(1)=SUREL
      I=1
      CALL IMPL7
      A(1,1)=DR1DU
      A(1,2)=DR1DT
      A(1,3)=DR1DV
      A(2,1)=DR2DU
      A(2,2)=DR2DT
      A(2,3)=DR2DV
      C(1)=DR1DU*U+DR1DV*V+DR1DT*R-DR1
      C(2)=DR2DU*U+DR2DV*V+DR2DT*R-DR2
      RETURN
      END
      SUBROUTINE  IMPL4
      COMMON    Z(150),DIST(150),YIN(150),VIN(150),EV(99),EY(99),
     1   C(31),DV(150),DY(150),A(31,31),C1(31),STAGE(150),EH(99),
     2   DG(150),DISCH(150),T(150),VT(160),YT(160),Q(160),EQ(160),
     3   X(150),EA(150)
      COMMON   G,DTIME,SC,DTDX,DXDT,DX,DT,TIME,DR1,DR2,S,R,V,U,
     1   DR1DU,DR1DV,DR1DS,DR1DT,DR2DU,DR2DV,DR2DS,DR2DT
      COMMON NN,N,L,NL,MN,I,MM
C     COMPUTATIONS  AT INTERIOR POINTS
  100 DO 150 I=2,NN
      CALL IMPL7
      J=2.*I-2.
      M=2.*I-1
      A(M,J)=DR1DS
      A(M,J+1)=DR1DU
      A(M,J+2)=DR1DT
      A(M,J+3)=DR1DV
      A(M+1,J)=DR2DS
      A(M+1,J+1)=DR2DU
      A(M+1,J+2)=DR2DT
      A(M+1,J+3)=DR2DV
```

```
      C(M)=DR1DU*U+DR1DV*V+DR1DT*R-DR1+DR1DS*S
      C(M+1)=DR2DU*U+DR2DV*V+DR2DT*R-DR2+DR2DS*S
  150 CONTINUE
      RETURN
      END
      SUBROUTINE  IMPL5
      COMMON    Z(150),DIST(150),YIN(150),VIN(150),EV(99),EY(99),
     1   C(31),DV(150),DY(150),A(31,31),C1(31),STAGE(150),EH(99),
     2   DG(150),DISCH(150),T(150),VT(160),YT(160),Q(160),EQ(160),
     3   X(150),EA(150)
      COMMON   G,DTIME,SC,DTDX,DXDT,DX,DT,TIME,DR1,DR2,S,R,V,U,
     1   DR1DU,DR1DV,DR1DS,DR1DT,DR2DU,DR2DV,DR2DS,DR2DT
      COMMON NN,N,L,NL,MN,I,MM
C
C     COMPUTATIONS AT  DOWNSTREAM  BOUNDARY
      P=EV(N)
      R=EY(N)
      DADY=DAY(R,N)
      D1=0.498982E-16
      D2=0.443349E-11
      D3=0.146063E-06
      D4=0.247673E-02
      D5=1.75506
      V1=P*P
      A1=AR(R,N)
      A2=A1*A1
      W=P*A1
      DR1=R+D1*W**4-D2*W**3+D3*W*W-D4*W-D5
      DR1DT=1.+DADY*P*(4.*D1*V1*P*A1*A2-3.*D2*V1*A2+2.*D3*P*A1-D4)
      DR1DV=A1*(D1*A2*A1*4.*V1*P-3.*D2*V1*A2+2.*D3*A1*P-D4)
      A(NL,NL-1)=DR1DT
      A(NL,NL)=DR1DV
      C(NL)=DR1DT*R+DR1DV*P-DR1
      RETURN
      END
      SUBROUTINE  IMPL7
      COMMON    Z(150),DIST(150),YIN(150),VIN(150),EV(99),EY(99),
     1   C(31),DV(150),DY(150),A(31,31),C1(31),STAGE(150),EH(99),
     2   DG(150),DISCH(150),T(150),VT(160),YT(160),Q(160),EQ(160),
     3   X(150),EA(150)
      COMMON   G,DTIME,SC,DTDX,DXDT,DX,DT,TIME,DR1,DR2,S,R,V,U,
     1   DR1DU,DR1DV,DR1DS,DR1DT,DR2DU,DR2DV,DR2DS,DR2DT
      COMMON NN,N,L,NL,MN,I,MM
C
C   COMPUTATIONS OF COEFFICIENT MATRIX AND COLUMN OF RESIDUES
C   SOLUTION OF SYSTEM OF EQUATIONS WITH FOUR NONZERO TERMS ON
C     THE  BAND  MATRIX
C
      U=EV(I)
      V=EV(I+1)
      S=EY(I)
      R=EY(I+1)
      SB=YIN(I)
      RB=YIN(I+1)
      UB=VIN(I)
      VB=VIN(I+1)
```

```
A1=AR(S,I)
A2=AR(R,I+1)
AB1=AR(SB,I)
AB2=AR(RB,I+1)
B1=BR(S,I)
B2=BR(R,I+1)
BB1=BR(SB,I)
BB2=BR(RB,I+1)
CB1=CI(SB)+(CN(SB)-CI(SB))*DG(I)
CB2=CI(RB)+(CN(RB)-CI(RB))*DG(I+1)
C01=CI(S)+(CN(S)-CI(S))*DG(I)
C02=CI(R)+(CN(R)-CI(R))*DG(I+1)
Q1=Q(I)
Q2=Q(I+1)
DB1DY=DBY(S,I)
DB2DY=DBY(R,I+1)
DB3DY=DDAY(S,I)
DB4DY=DDAY(R,I+1)
DA1DY=DAY(S,I)
DA2DY=DAY(R,I+1)
DA3DY=DAY(SB,I)
DA4DY=DAY(RB,I+1)
Z1=Z(I)
Z2=Z(I+1)
S0=(Z1-Z2)/DX
V1=V+VB-U-UB
V2=V+U-VB-UB
D1=R-S+RB-SB
D2=R+S-SB-RB
VD=V+U+VB+UB
E1=DA1DY
E2=DA2DY
EB1=DA3DY
EB2=DA4DY
QD=Q1*(U/A1+V/A2+UB/AB1+VB/AB2)
BD=1./E1+1./E2+1./EB1+1./EB2
AD=A1/E1+A2/E2+AB1/EB1+AB2/EB2
SF1=C01*C01*U*ABS(U)*(B1/A1)**(4./3.)/2.2082
SF2=C02*C02*V*ABS(V)*(B2/A2)**(4./3.)/2.2082
SF3=CB1*CB1*UB*ABS(UB)*(BB1/AB1)**(4./3.)/2.2082
SF4=CB2*CB2*VB*ABS(VB)*(BB2/AB2)**(4./3.)/2.2082
DSF2=C02*C02*V*(B2/A2)**(4./3.)/2.2082
DSF1=C01*C01*U*(B1/A1)**(4./3.)/2.2082
SF=SF1+SF2+SF3+SF4
DB4=DB1DY/B1
DB3=DB3DY/(E1*E1)
DB2=DB4DY/(E2*E2)
DB1=DB2DY/B2
DA4=DA1DY/A1
DA3=DA1DY/(A1*A1)
DA2=DA2DY/(A2*A2)
DA1=DA2DY/A2
DR1=.25*DTDX*VD*D1+C2+DTDX*.25*V1*AD-.5*Q1*DT*BD
DR2=DXDT*V2/G+.25*VD*V1/G+.5*DX*QD/G+D1-DX*2.*S0+.5*DX*SF
W=.25*DTDX
DR1DS=1.-W*VD+V1*(DA1DY/E1-A1*DB3)*W+.5*DT*DB3*Q1
```

```
      DR1DT=W*VD+1.+W*V1*(CA2CY/E2-A2*DB2)+.5*DT*Q1*DB2
      DR1DV=.25*DTDX*D1+DTCX*AD*.25
      DR1DU=.25*DTDX*D1-DTCX*AD*.25
      DR2DU=(DXDT-.5*(U+UB)+.5*DX*Q1/A1)/G+DX*DSF1
      DR2DV=(DXDT+.5*(V+VB)+.5*DX*Q1/A2)/G+DX*DSF2
      DR2DT=1.-.5*DX*Q1*V*CA2/G+CX*(4./3.)*SF2*(DB1-DA1)*.5
      DR2DS=-1.-.5*DX*Q1*U*CA3/G+DX*(4./3.)*SF1*(DB4-DA4)*.5
      RETURN
      END
      SUBROUTINE MSIMQ (A,B,N,KS)
      DIMENSION A(1),B(1)
C
C             FORWARD  SOLUTION
      TOL=0.0
      KS=0
      JJ=-N
      DO 65 J=1,N
      JY=J+1
      JJ=JJ+N+1
      BIGA=0.
      IT=JJ-J
      L1=J+4
      IF(L1-N) 10,10,12
   12 L1=N
   10 DO 30 I=J,L1
C
C          SEARCH FOR MAXIMUM COEFFICIENT IN COLUMN
C
      IJ=IT+I
      IF(ABS(BIGA)-ABS(A(IJ))) 20,30,30
   20 BIGA=A(IJ)
      IMAX=I
   30 CONTINUE
C
C        TEST FOR PIVOT LESS THAN TOLENANCE(SINGULAR MATRIX)
C
   32 IF(ABS(BIGA)-TOL) 35,35,40
   35 KS=1
      RETURN
C
C           INTERCHANGE ROWS IF NECESSARY
C
   40 I1=J+N*(J-2)
      IT=IMAX-J
      L2=J+4
      IF(L2-N) 42,42,44
   44 L2=N
   42 DO 50 K=J,L2
      I1=I1+N
      I2=I1+IT
      SAVE=A(I1)
      A(I1)=A(I2)
      A(I2)=SAVE
C
C          DIVIDE EQUATION BY LEADING COEFFICIENT
C
```

```
   50 A(I1)=A(I1)/BIGA
   52 SAVE=B(IMAX)
      B(IMAX)=B(J)
      B(J)=SAVE/BIGA
C
C           ELIMINATE NEXT VARIABLE
C
      IF(J-N) 55,70,55
   55 IQS=N*(J-1)
      L3=JY+4
      IF(L3-N) 57,57,59
   59 L3=N
   57 DO 65 IX=JY,L3
      IXJ=IQS+IX
      IT=J-IX
      DO 60 JX=JY,L3
      IXJX=N*(JX-1)+IX
      JJX=IXJX+IT
   60 A(IXJX)=A(IXJX)-(A(IXJ)*A(JJX))
   65 B(IX)=B(IX)-(B(J)*A(IXJ))
C
C           BACK  SOLUTION
C
   70 NY=N-1
      IT=N*N
      DO 80 J=1,NY
      IA=IT-J
      IB=N-J
      IC=N
      DO 80 K=1,J
      B(IB)=B(IB)-A(IA)*B(IC)
      IA=IA-N
   80 IC=IC-1
      RETURN
      END
      FUNCTION DDAY(Y,I)
      DIMENSION YE(3,16)
      DATA YE/ 1.81986,  -18.24,    89.7562,
     2          .06211704,19.56738, -80.5784,
     3          .04001556,24.41952, -132.7526,
     4          1.543296, -9.25536, 1.388864,
     5          2.286108, -31.9605, 112.9622,
     6          1.361472, -19.59354,103.2114,
     7          -.0282742,6.15426,  27.3412,
     8          -.04109472,13.46364,-12.46158,
     9          -.476388, 20.30124, -23.264,
     $          .6788832, -11.44002,161.6744,
     1          -3.265044,93.5712,  -372.274,
     2          -.2242944,23.7729,  -115.4414,
     3          .6326076, -.01256112,-18.28808,
     4          .6085212, -10.52872,127.5636,
     5          -1.861308,55.21506, -178.6174,
     6          -1.549296,55.34604, -234.55/
      DDAY=YE(1,I)*Y*Y+YE(2,I)*Y+YE(3,I)
      RETURN
      END
```

```
      FUNCTION DBY(Y,I)
      DIMENSION YD(4,16)
      DATA YD/ 0.0676796,  -1.117932,  16.92792,  -14.3062,
     2 -.1905932,  6.47532,  -38.199,  56.7726,
     3 -0.468548,  16.4217,    -151.6936,  407.595,
     4 -0.2995736, 12.77205, -136.3238,  422.67,
     5 -.02592076, 2.606121, -27.9424,   81.0191,
     6 0.0701952,  -1.352286, 11.162,    7.38096,
     7 0.02232208, -0.686499, 11.29932,  18.5439,
     8-0.154314,   5.40538,  -42.1238,   140.816,
     9 -0.0915504, 4.13886,  -38.766,    167.453,
     $ 0.03260408, -0.627045, 3.71032,   115.951,
     1 -0.1009396,  4.32945, -34.1462, 82.0455,
     2 -0.1040672, 3.54735,  -20.642,   21.1131,
     3 -.1035248,  4.24338,  -36.4832,   80.5002,
     4 0.03616684, -0.876678,7.12684,    72.3461,
     5 0.1517292,  -7.76796, 123.7250,   -453.022,
     6 -0.0980716, 2.52573,  3.4066,     -89.3602/
      DBY=YD(1,I)*Y*Y*Y+YD(2,I)*Y*Y+YD(3,I)*Y+YD(4,I)
      RETURN
      END
      FUNCTION BR(Y,I)
      DIMENSION YC(5,16)
      DATA YC/ .0169199,  -0.372644, 8.46396,  -14.3062,   156.411,
     2 -0.0476483,  2.15844,  -19.0995,   56.7726,  135.743,
     3 -0.117137,  5.47390,    -75.8468,   407.595,  -521.795,
     4 -0.0748934, 4.25735,  -68.1619,   422.671,  -568.138,
     5 -0.00648019,0.868707,   -13.9712, 81.0191,  107.281,
     6 0.0175488,  -0.450762,  5.581,    7.38096,  170.019,
     70.00558052,  -0.223833,  5.64966,  18.5439,  175.062,
     8 -0.0385785, 1.80196,    -21.0619, 140.816,  111.684,
     9 -0.0228876, 1.37962,  -19.383,    167.453,  110.199,
     $ 0.00815102, -0.209015,  1.85516,  115.951,  154.635,
     1 -0.0252349, 1.44315,  -17.0731,  82.0455,   98.1520,
     2 -0.0260168, 1.18245,    -10.0321,  21.1131, 237.568,
     3 -0.0258812,  1.41446,  -18.2416,   80.5002,  144.160,
     4 .00904171,  -0.292226,  3.56342,  72.3461,  170.58,
     5 0.0379323,  -2.58932,   61.8625,  -453.022, 1244.60,
     6 -0.0245179, 0.84191,  1.70330,  -89.3602,   548.324/
      BR=YC(1,I)*Y**4+YC(2,I)*Y**3+YC(3,I)*Y*Y+YC(4,I)*Y+YC(5,I)
      RETURN
      END
      FUNCTION AR(Y,I)
      DIMENSION YA(5,16)
      DATA YA/ .151655,  -3.04,    44.8781,  -.767774, 53.3734,
     2         .00517641,3.261238, -40.2892, 300.98,   -29.4285,
     3         .00333463,4.06992,  -66.3763, 439.725,  -90.4865,
     4         .128608,  -1.54256, .694432,  287.073,  -76.2072,
     5         .190509,  -5.32675, 56.4811,  31.2948,  12.8542,
     6         .113456,  -3.26559, 51.6057,  25.3572,  41.2548,
     7         -.00235618,1.02571, 13.6706,  186.127,  3.95523,
     8         -.00342456,2.24394, -6.23079, 337.834,  -59.6666,
     9         -.039699, 3.38354,  -11.632,  418.848,  -102.304,
     $         .0565736, -1.90667, 80.8372,  101.179,  14.7159,
     1         -.272087, 15.5552,  -186.137, 831.848,  -158.55,
     2         -.0186912,3.96215,  -57.7207, 429.661,  -76.3123,
```

```
3          .0527173, -.00209352,-9.14404,276.019,  -38.9704,
4          .0507101, -1.75479, 63.7818,  99.292,   31.1864,
5          -.155109, 9.20251,  -89.3087, 452.143,  -37.2568,
6          -.129108, 9.22434,  -117.275, 625.933,  -109.43/
 AR=YA(1,I)*Y**4+YA(2,I)*Y**3+YA(3,I)*Y*Y+YA(4,I)*Y+YA(5,I)
 RETURN
 END
 FUNCTION DAY(Y,I)
 DIMENSION YB(4,16)
 DATA YB/ .60662,   -9.12,    89.7562,  -.767774,
2          .02070568,9.78369,  -80.5784, 300.98,
3          .01333852,12.20976,-132.7526,439.725,
4          .514432,  -4.62768, 1.388864, 287.073,
5          .762036,  -15.98025,112.9622, 31.2948,
6          .453824,  -9.79677, 103.2114, 25.3572,
7          -.00942472,3.07713, 27.3412,  186.127,
8          -.01369824,6.73182, -12.46158,337.834,
9          -.158796, 10.15062, -23.264,  418.848,
$          .2262944, -5.72001, 161.6744, 101.179,
1          -1.088348,46.7856,  -372.274, 831.848,
2          -.0747648,11.88645, -115.4414,429.661,
3          .2108692, -.00628056,-18.28808,276.019,
4          .2028404, -5.26437, 127.5636, 99.292,
5          -.620436, 27.60753, -178.6174,452.143,
6          -.516432, 27.67302, -234.55,  625.933/
 DAY=YB(1,I)*Y**3+YB(2,I)*Y*Y+YB(3,I)*Y+YB(4,I)
 RETURN
 END
 FUNCTION CI(Y)
 CI=((((.281877E-06*Y-.24522E-04)*Y+.000762819)*Y-.0102767)*Y+
10.0614686)*Y-0.0876877                                              CI
 RETURN
 END
 FUNCTION CN(Y)
 CN=(((.187844E-5*Y-.000119336)*Y+.00264335)*Y-.02147)*Y+.093644
 RETURN
 END
```