US Army Corps of Engineers

WATER OPERATIONS TECHNICAL SUPPORT PROGRAM

INSTRUCTION REPORT E-90-1

CE-QUAL-RIV1: A DYNAMIC, ONE-DIMENSIONAL (LONGITUDINAL) WATER QUALITY MODEL FOR STREAMS

USER'S MANUAL

Environmental Laboratory

DEPARTMENT OF THE ARMY
Waterways Experiment Station, Corps of Engineers
3909 Halls Ferry Road, Vicksburg, MS 39180-6199

November 1990
Final Report

Prepared for DEPARTMENT OF THE ARMY
US Army Corps of Engineers
Washington, DC 20314-1000

Under CWIS Work Unit 31595

Destroy this report when no longer needed. Do not return it to the originator.

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

The contents of this report are not to be used for advertising, publication, or promotional purposes. Citation of trade names does not constitute an official endorsement or approval of the use of such commercial products.

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY Unclassified	3 DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) Instruction Report E-90-1	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION USAEWES Environmental Laboratory	6b. OFFICE SYMBOL (If applicable)
6c. ADDRESS (City, State, and ZIP Code) 3909 Halls Ferry Road Vicksburg, MS 39180-6199	7a. NAME OF MONITORING ORGANIZATION 7b. ADDRESS (City, State, and ZIP Code)
8a. NAME OF FUNDING/SPONSORING ORGANIZATION US Army Corps of Engineers	8b. OFFICE SYMBOL (If applicable)
8c. ADDRESS (City, State, and ZIP Code) Washington, DC 20314-1000	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER 10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO. CWIS 31595

11. TITLE (Include Security Classification)
CE-QUAL-RIV1: A Dynamic, One-Dimensional (Longitudinal) Water Quality Model for Streams, User's Manual

12. PERSONAL AUTHOR(S)

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final report	FROM _____ TO _____	November 1990	249

16. SUPPLEMENTARY NOTATION
Available from National Technical Information Service, 5285 Port Royal Road, Springfield, VA 22161.

17. COSATI CODES		
FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
Dynamic
One-dimensional
Streams
Water quality model

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

A dynamic, one-dimensional (longitudinal), water quality model for unsteady flows in rivers and streams, CE-QUAL-RIV1, is presented.

CE-QUAL-RIV1 is developed in two parts, hydrodynamic and water quality. Output from the hydrodynamic solution is used to drive the water quality model.

The hydrodynamic code uses a four-point implicit Newton-Raphson procedure to solve the nonlinear St. Venant equation. Numerical accuracy for the advection of sharp gradients is preserved in the water quality code through the use of the explicit two-point, fourth-order accurate, Holly-Preissmann scheme.

(Continued)

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT	21 ABSTRACT SECURITY CLASSIFICATION
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	Unclassified
22a. NAME OF RESPONSIBLE INDIVIDUAL	22b TELEPHONE (Include Area Code) / 22c. OFFICE SYMBOL

DD Form 1473, JUN 86 *Previous editions are obsolete* SECURITY CLASSIFICATION OF THIS PAGE

Unclassified

19. ABSTRACT (Continued).

Water quality constituents include temperature, dissolved oxygen, carbonaceous biochemical oxygen demand, organic nitrogen, ammonia nitrogen, nitrate nitrogen, orthophosphate phosphorus, coliform bacteria, dissolved iron, and dissolved manganese. The effects of algae and macrophytes are also included.

The model allows simulation of branched river systems with multiple hydraulic control structures, such as run-of-the-river dams, waterway locks and dams, and reregulation dams. The model was developed to simulate the transient water quality conditions associated with highly unsteady flows that can occur on regulated streams.

PREFACE

The water quality model documented by this instruction report was sponsored by Headquarters, US Army Corps of Engineers (HQUSACE), as a part of the Environmental and Water Quality Operational Studies, Work Unit IC.3 (CWIS Work Unit 31595), entitled "Improve and Verify Riverine Water Quality and Ecological Predictive Techniques." The HQUSACE Technical Monitors were Mr. Earl Eiker, Dr. John Bushman, and Mr. James Gottesman.

This report serves as the user's manual for the dynamic riverine water quality model, CE-QUAL-RIV1. The first draft of this manual, dated September 1982, was prepared by Drs. Keith W. Bedford, Robert M. Sykes, and Charles Libicki of Ohio State University under Contract No. DACW39-82-3548. This present version of the user's manual is a result of revisions to the 1982 draft user's manual and reflects model modifications made by Ohio State University after 1982 and by the Environmental Laboratory (EL), US Army Engineer Waterways Experiment Station (WES), Vicksburg, MS. The manual revisions were made by Dr. Mark Dortch and Ms. Toni Schneider of the Water Quality Modeling Group (WQMG), Ecosystem Research and Simulation Division (ERSD), EL. The WES revisions to the model code were made by Dr. Dortch, Ms. Schneider, Dr. James Martin, and Dr. Marc Zimmerman of the WQMG, and Dr. D. M. Griffin, who was working through an Interagency Personnel Agreement with Louisiana Tech University. This work was conducted under the general supervision of Dr. John Harrison, Chief, EL, and Mr. Donald Robey, Chief, ERSD, and under the direct supervision of Dr. Dortch, Chief, WQMG.

This report was published under the Water Operations Technical Support (WOTS) Program managed under the Environmental Resources Research and Assistance Programs (ERRAP), EL. Mr. J. Lewis Decell is the manager, ERRAP.

This manual was reviewed by Dr. Zimmerman and Ms. Schneider of the WQMG. The report was edited by Ms. Lee T. Byrne of the WES Information Technology Laboratory.

Figure 6 is used courtesy of the American Society of Civil Engineers, Figures 7 and 8 courtesy of the Water Pollution Control Federation, and Table 2 courtesy of John Wiley and Sons, Inc.

Commander and Director of WES during publication of this report was COL Larry B. Fulton, EN. Dr. Robert W. Whalin was Technical Director.

This report should be cited as follows:

Environmental Laboratory. 1990. "CE-QUAL-RIV1: A Dynamic, One-Dimensional (Longitudinal) Water Quality Model for Streams: User's Manual," Instruction Report E-90-1, US Army Engineer Waterways Experiment Station, Vicksburg, MS.

CONTENTS

LIST OF FIGURES

CONVERSION FACTORS, NON-SI TO SI (METRIC)
UNITS OF MEASUREMENT

Non-SI units of measurement used in this report can be converted to SI (metric) units as follows:

Multiply	By	To Obtain
cubic feet	0.02831685	cubic metres
feet	0.3048	metres
inches	25.4	millimetres
miles (US statute)	1.609347	kilometres
square feet	0.09290304	square metres

CE-QUAL-RIV1: A DYNAMIC, ONE-DIMENSIONAL (LONGITUDINAL) WATER QUALITY MODEL FOR STREAMS

USER'S MANUAL

PART I: INTRODUCTION

Background

1. The US Army Corps of Engineers (CE) has major responsibility for the regulation of the Nation's streams, rivers, and waterways. This activity often involves resolving issues and concerns with regard to the water quality of these regulated systems. Water quality simulation models can be powerful tools for studying these issues. However, to be useful, the water quality model must be properly suited for the problem at hand.

2. Regulated stream systems may include complicating physical features, such as multiple run-of-the-river dams, locks and dams, and reregulation dams. Additionally, highly unsteady flows may exist or may be in the planning, as with peaking hydropower releases. There are numerous water quality models in existence, but most were developed for steady flow conditions and are not appropriate when time-varying flows are to be considered.

3. The model presented herein was originally developed at Ohio State University for the Environmental Protection Agency (EPA). The purpose of the development was for predicting water quality associated with storm water run-off. Researchers at the US Army Engineer Waterways Experiment Station (WES) were attracted to the model because it is fully dynamic for flow and water quality and it has several desirable numerical features, such as the two-point fourth-order scheme for accurately advecting water quality concentrations. The WES contracted Ohio State University to modify the code to handle control structures. This modification, along with the unsteady flow feature, gave the model the versatility needed for simulating CE-regulated stream/waterway projects. Subsequently, the finished version was tested at WES, and additional modifications and corrections were made, resulting in the model presented herein, CE-QUAL-RIV1.

Objective

4. The purpose of this manual is to document the mathematical and numerical formulations of CE-QUAL-RIV1 and to provide the guidance necessary for its implementation. The first six parts deal with documentation, whereas the last two parts are for operation. Some parts are associated only with the hydraulic code, RIV1H, whereas others deal only with the water quality code, RIV1Q. This is due to the fact that the codes are separate. The output from RIV1H is used to drive RIV1Q. It would also be possible to use another hydraulic code, similar to RIV1H, to drive RIV1Q.

Model Selection Criteria

5. The first criterion for selection of CE-QUAL-RIV1 for an application is whether the issues can be resolved with a one-dimensional (1-D) (cross sectionally averaged) model. Most riverine water quality issues can be resolved with a 1-D model; that is, lateral and vertical gradients in water quality constituent concentrations are insignificant and unimportant relative to longitudinal gradients. This is generally true in riverine systems. This assumption implicitly means that vertical temperature, density, and chemical stratifications (which can play a dominant role in the water quality of lakes and reservoirs) are nonexistent or negligible for practical purposes. Thus, although this model can be used for run-of-the-river reservoirs, locks and dams, and reregulation pools, the user must first be sure that vertical stratification does not exist or is so minor as to not affect water quality conditions. Guidance for determining the stratification potential of impounded streams can be found in Ford and Johnson (1986), Appendix A.

6. The second criterion for selection of CE-QUAL-RIV1 has to do with the nature of the flow and the issues; although CE-QUAL-RIV1 was developed for water quality simulations of riverine systems with highly unsteady flow, it can be used for steady flow conditions. However, it may be easier and more economical to use another more simplistic formulation, such as the EPA QUAL2E model (Brown and Barnwell 1987), which assumes steady flow. If the issues demand high resolution and the flows change substantially over a period of hours or days, then a fully dynamic model, such as CE-QUAL-RIV1 should be used.

7. Specific guidance has not been developed to determine at what condition a dynamic flow model should be used. The development of specific guidance would depend on the issues and the required resolution. For example, if daily average (or longer time averages, perhaps even steady-state) predictions are sufficient, then more simplistic modeling approaches may suffice, even for unsteady flow projects. However, if diel fluctuations (and peaks and troughs) are important and various interest groups are sensitive to such fluctuations, then a dynamic model would be necessary for an unsteady flow project. Transient flow conditions can produce substantially greater fluctuations in diel temperature and dissolved oxygen (DO) values than the natural diel effects (Matter et al. 1983).

8. Even when a study is highly sensitive and high resolution is required with diel fluctuations, a time-varying water quality model that assumes steady flow (such as QUAL2E) may suffice if the flows are relatively constant or change slowly during the simulation with respect to the travel time of the system. However, if the flows change substantially during a day and from day to day, then a dynamic flow model (such as CE-QUAL-RIV1) should be considered. The study of riverine water quality resulting from the releases from peaking hydropower dams is a good example of the use of CE-QUAL-RIV1.

9. It should be noted that RIV1H uses the fully dynamic flow equations; thus it has wide-ranging capabilities with good resolution. This advantage must be weighed against possible disadvantages such as familiarity of the user with the code. RIV1Q can be driven by any hydraulic or hydrologic routing model, no matter how simple, as long as the proper linkages are made.

PART II: THE GOVERNING EQUATIONS

General Considerations

10. The transport of momentum and water quality constituents during unsteady flows can be marked by sharp gradients in flow, elevation, and water quality concentrations. These gradients can be propagated by the flow wave through regions of highly variable cross section intermittently joined by major inflow tributaries, with the magnitudes of concentrations often being augmented by nonpoint source contributions.

11. Since the speed of flow waves is often quite high, water quality concentrations can be dominated by direct transport advection rather than biogeochemical interactions and diffusion. Therefore, the water quality model must be dynamic and have the minimum following attributes:

a. Account for time-varying flow, elevation, and water quality constituent changes resulting from highly unsteady flows.

b. Include the direct explicit interaction of flow and elevation on the constituent distributions.

c. Be applicable for a river channel of arbitrary cross section and specified bottom slope.

d. Allow for a number of water quality constituents and the proper mathematical specification of their mathematical interrelationship.

e. Account for the effects of lateral inputs of water and associated pollutant concentrations.

f. Allow simulation of multiple hydraulic control structures.

12. The following section lists the assumptions that the above attributes require for the development of the basic governing transport equations.

13. For a river that is much longer than wide or deep, the following assumptions apply:

a. Hydrostatic pressure is assumed.

b. Lateral and vertical gradients are small and neglected; thus the equations are cross sectionally averaged for flow and constituent variables (1-D assumption).

c. All cross sections and bottom configurations are known.

d. All lateral point and nonpoint source flows and input concentrations are known.

14. When solved, the hydraulic transport equations permit the calculation of downstream histories of flow and water surface elevation. These equations have been known for some time, and their derivation is quite routine. There are two different approaches to the derivation of flow wave equations. The first starts with the basic three-dimensional (3-D) equations of continuity and Navier Stokes (Bird, Stewart, and Lightfoot 1964) and by cross-sectional averaging reduces the four equations to a coupled pair of dynamic cross sectionally averaged equations for longitudinal space and time patterns of flow and elevation. This procedure, however, always results in the necessity of specifying a very ambiguous eddy viscosity to account for the ever-present correlations between fluctuating components resulting from the averaging. Longitudinal eddy viscosities are very small, particularly during elevated flows (Fischer et al. 1979), and are quite frequently used improperly to tune the correct answer into existence rather than to represent actual problem physics. The control volume method (Liggett 1975) is used to derive the equations herein.

15. The notation and coordinate system is defined as in Figure 1. It is assumed that the coordinate system is placed in the river bottom with the bed slope relative to a gravity-based coordinate system being such that $\tan \theta \simeq \theta$, and, therefore, θ is very small. It is further assumed that x is directed positive downstream and that because the bed slope is very small, the water elevation $h(x,t)$, directed parallel to the gravitational direction, is perpendicular to x. $A(x,t)$ is the cross-sectional area, and $B(x,t)$ is the channel top width; by knowing the shape factor $\xi(z)$, A and B can be related to $h(x,t)$.

16. Assume as in Figure 2 that a discrete length, Δx, of river channel is isolated. If the flow is from left to right, then unit normals n_o and n_i are defined as being positive away from each face through which flow is entering and exiting. Total mass, momentum, and species mass will be conserved within this control volume. The general form of the control volume conservation law for a continuum of mass concentration b is (Streeter and Wylie 1979)

$$\frac{d\beta}{dt} = \iiint_{CV} \frac{\partial b}{\partial t} \, dV + \iint_{CS} b(\bar{v} \cdot \hat{n}) \, dA \tag{1}$$

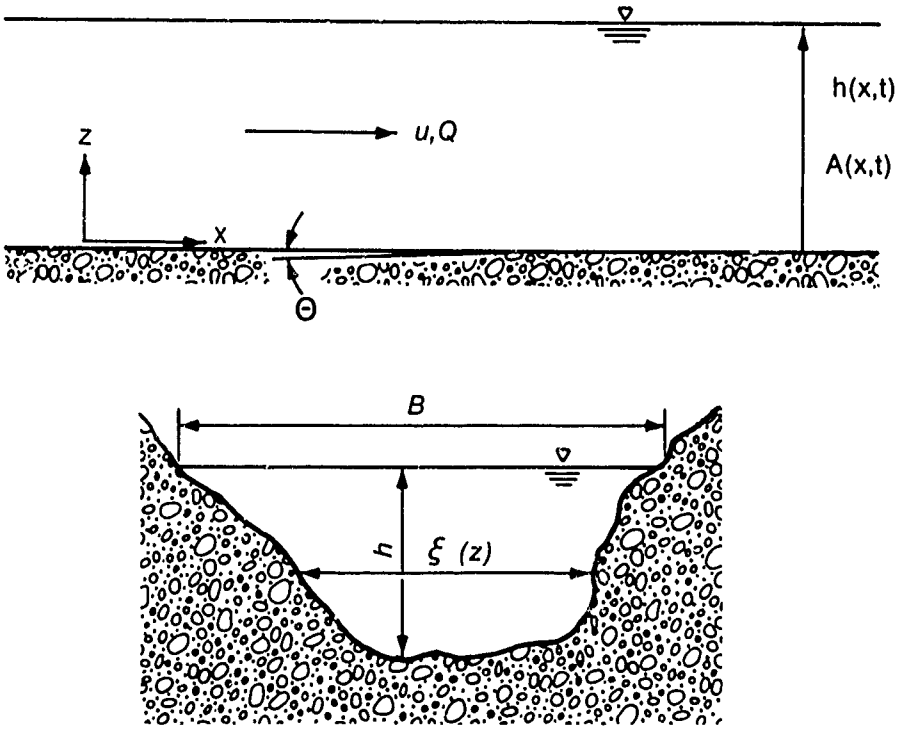


Figure 1. Coordinate system and notation

Equation 1 states that the total time rate at which the mass in the control volume, β , ($\beta = b\Psi$) changes equals the time rate of accumulation of b within the control volume (cv) plus the net rate at which b is being carried into the control volume through the control surface (cs). The control volume conservations of water mass, momentum, and species mass are respectively written for an arbitrary control volume in a 3-D flow field $\bar{v}$ as

$$\underline{\text{mass}}: \iiint_{cv} \frac{\partial \rho}{\partial t} \, d\Psi + \iint_{cs} (\rho \bar{v} \cdot \hat{n}) \, dA = 0 \tag{2}$$

$$\underline{\text{momentum}}: \iiint_{cv} \frac{\partial \bar{v} \rho}{\partial t} \, d\Psi + \iint_{cs} \bar{v} (\rho \bar{v} \cdot \hat{n}) \, dA = \bar{F} \tag{3}$$

$$\underline{\text{mass species}}: \iiint_{cv} \frac{\partial \alpha}{\partial t} \, d\Psi + \iint_{cs} \alpha (\bar{v} \cdot \hat{n}) \, dA = S^* \tag{4}$$

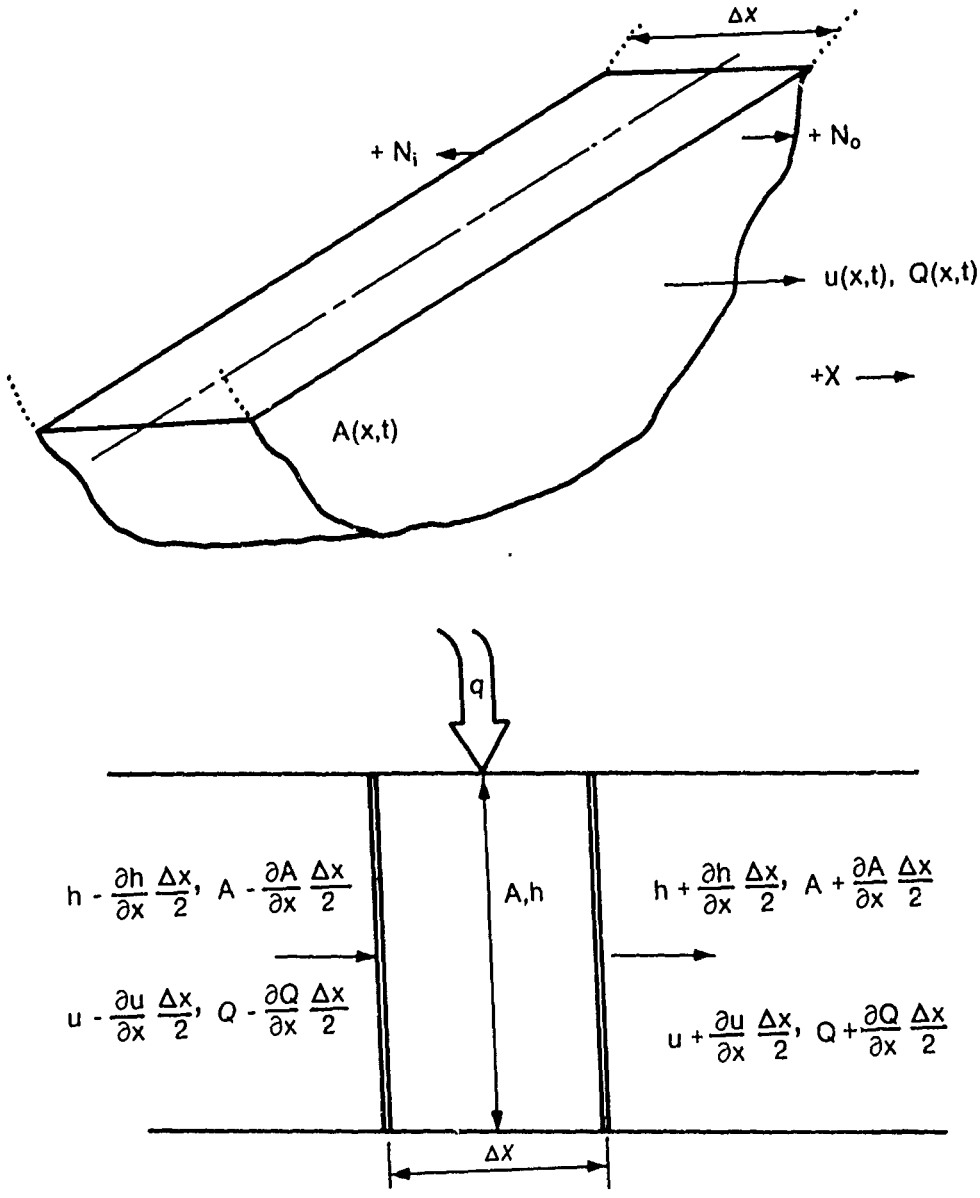


Figure 2. Control volume definitions

where

ρ = fluid density, M/L^3

t = time

$d\forall$ = differential volume, L^3

$\bar{v}$ = velocity vector $(u\hat{i} + v\hat{j} + w\hat{k})$, L/t

dA = differential area, L^2

$\bar{F}$ = vector sum of the real applied external forces on the cv , ML/t^2

α = species mass concentration, M/L^3

S^* = net source term for biochemical changes in α , M/t

The variables M , L , and t are general units of mass, length, and time, respectively.

Derivation of Equations

17. As in Liggett (1975), a much simpler derivation is obtained for the river problem by assuming that the average velocity, U , (or average species concentration α) is known at a cross section. A Taylor series expansion across inlet and outlet permits much simpler expressions to be identified.

Conservation of mass

18. The conservation of water mass equation says the time rate of change of storage equals net mass efflux through the control volume surface. By a Taylor series expansion, therefore

$$\frac{\partial(\rho A \Delta x)}{\partial t} = \rho\left[\left(U - \frac{\partial U}{\partial x}\frac{\Delta x}{2}\right)\left(A - \frac{\partial A}{\partial x}\frac{\Delta x}{2}\right) - \left(U + \frac{\partial U}{\partial x}\frac{\Delta x}{2}\right)\left(A + \frac{\partial A}{\partial x}\frac{\Delta x}{2}\right)\right] \tag{5}$$

For an incompressible fluid such as water, ρ = constant ; multiplying out and dividing by $\rho\Delta x$ gives for small Δx

$$\frac{\partial A}{\partial t} + \frac{\partial (UA)}{\partial x} = 0 \quad \text{or} \quad \frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} = 0 \tag{6}$$

where Q is the volume flow rate.

Momentum conservation

19. The momentum equation is a bit more difficult. It states that the net force acting on the cv equals the time rate of change of momentum in the

cv plus the net rate of efflux of momentum through the cv . Therefore

$$\frac{\partial(Q\Delta x)\rho}{\partial t} + \rho\left\{\left[U(UA)\right] + \frac{\partial}{\partial x}\left[U(UA)\right]\frac{\Delta x}{2}\right\}$$

$$-\rho\left\{\left[U(UA)\right] - \frac{\partial}{\partial x}\left[U(UA)\right]\frac{\Delta x}{2}\right\} = \bar{F} \qquad (7)$$

The force vector $\bar{F}$ requires further expansion into three subcategories, gravity, shear, and pressure forces.

20. Gravity force, f_g . The total gravity force is nothing more than the component of the weight of water in the control volume ($\rho g A \Delta x$) directed in the x-direction or $\rho g A \Delta x \sin\theta$. Therefore,

$$f_g = \rho g A \Delta x S_o \qquad (8)$$

where $S_o = \sin\theta \simeq \theta$ is the slope.

21. Shear force, f_τ . The channel sides and bottoms deplete momentum by the action of bottom friction or shear. This depletion is made mathematically analogous to the gravity slope term by setting

$$f_\tau = \rho g A \Delta x S_f \qquad (9)$$

where S_f is the friction slope. Several forms for the friction slope exist, but either Chezy or Manning forms predominate, i.e.

$$S_f = \frac{U^2 n^2}{\left[1.486\ R^{2/3}\right]^2} \qquad (10)$$

where

n = Manning friction factor

R = hydraulic radius which is approximately equal to A/B , where B is the top width

Therefore,

$$f_\tau = \rho g A \Delta x \frac{U^2 n^2}{\left[1.486\ R^{2/3}\right]^2}$$

or

$$f_\tau = \rho g A \Delta x \frac{n^2 |Q| Q}{2.2\ A^2\ R^{4/3}} \tag{11}$$

where the absolute value has been retained to ensure that no matter which way the wave propagates, shear always dissipates momentum.

22. Pressure force, f_p. The total pressure force on the face of the control volume is the integral of the irregular trace of the cv, i.e.

$$f_p = \int_0^h \rho g (h - z) \xi(z) dz \tag{12}$$

where $\xi(z)$ is the channel width at height z above the bottom. A Taylor series expansion gives the net pressure in the downstream direction

$$f_p = - \frac{\partial}{\partial x} \int_0^h \rho g (h - z) \xi(z) dz \Delta x$$

$$= - \rho g \int_0^h \frac{\partial}{\partial x} \left[(h - z)\ \xi(z)\right]\ dz \Delta x \tag{13}$$

and by chain rule differentiation

$$f_p = - \rho g \left[\frac{\partial h}{\partial x} \int_0^h \xi(z) dz + \int_0^h (h - z) \frac{\partial \xi(z)}{\partial x}\ dz \right] \Delta x \tag{14}$$

The first term represents the pressure force at that particular cross section. The second term represents the net pressure force caused by rapid area changes over the length Δx. If the channels are considered prismatic and regular, then the last term has little significance. Therefore

$$f_p = - \rho g A \frac{\partial h}{\partial x} \Delta x \tag{15}$$

The final equation for momentum is then

$$\frac{\partial Q}{\partial t} + \frac{\partial}{\partial x} (QU) + gA \frac{\partial h}{\partial x} = gA(S_o - S_f) \tag{16}$$

Modifications to momentum and continuity

23. Lateral and tributary inflow. Runoff from lands adjacent to the channel or tributary inflow can cause increased levels of total mass and momentum in the river. If q is the flow per unit channel length entering the river with velocity U_q, then Equations 6 and 16 become, respectively,

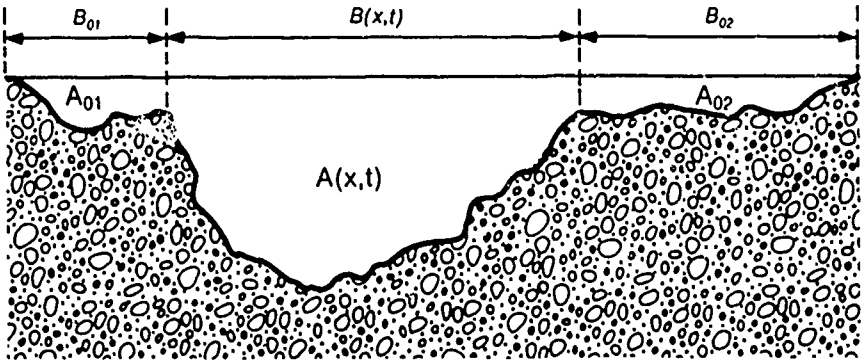
$$\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} = q \tag{17}$$

and

$$\frac{\partial Q}{\partial t} + \frac{\partial}{\partial x} (UQ) + gA \frac{\partial h}{\partial x} = gA(S_o - S_f) + qU_q \tag{18}$$

However, the model code does not include the last term in Equation 18 because it is relatively insignificant compared with other terms of the momentum equation.

24. Flood plain storage. As sometimes occurs, excess quantities of nonmoving water are often stored in the floodplain. Since the water is not moving, the momentum equation remains unaffected by this feature. The continuity equation must, however, account for the excess mass. If, as in Figure 3, A_o is defined as the cross-sectional area of the floodplain waters, then Equation 17 becomes



$$A_o = A_{01} + A_{02}$$
$$B_o = B_{01} + B_{02}$$

Figure 3. Floodplain geometry and notation

$$\frac{\partial (A + A_o)}{\partial t} + \frac{\partial Q}{\partial x} = q \tag{19}$$

However, the effect of floodplain storage is not presently included in the code.

25. Channel constrictions. Often very intense channel constrictions, due to bridges for example, occur over channel lengths that are far too small to economically resolve in the model. The subgrid scale effect of such constrictions is a momentum loss and backwater effect. This effect is accounted for in the right side of the momentum equation by subtracting the force term, $\rho g \Delta h_E$ where h_E is a head loss and K_E is a coefficient to be selected and optimized.

$$h_E = \frac{K_E}{2g}\left(\frac{Q}{A}\right)^2 \tag{20}$$

The default value is zero for no constriction loss. A value for K_E as high as 0.5 may be appropriate for an abrupt constriction.

26. Momentum correction factor. When the velocity across the channel is substantially nonuniform through the model reach, it may be necessary to use a momentum correction factor, β, in the momentum equation. The momentum

correction factor β multiplies the second term on the left side of Equation 18; this correction permits the use of the average velocity, U , in the solution whereas the velocity distribution at each cross section may be quite different from U . For example, for laminar flow in a straight round tube, β is 4/3. It equals 1.0 for uniform flow and cannot be less than 1.0. For RIV1H, a constant value is used throughout the modeled reach. The default value of 1.0 is recommended for rivers and streams.

27. Tributary networks. The momentum and continuity equations above must be applied to each and every tributary entering the main stem. At each junction, the water surfaces in each branch must be equal.

Mass species equation

28. Again, by using the Taylor series approach, the advection and diffusion of a biogeochemically reactive substance of concentration α is

$$\frac{\partial(\alpha A)}{\partial t} + \frac{\partial(Q\alpha)}{\partial x} = \frac{\partial}{\partial x}\left(DA\,\frac{\partial \alpha}{\partial x}\right) + \gamma q + S^* \qquad (21)$$

where

D = turbulent dispersion coefficient

γ = concentration of species type α entering the tributary from lateral flow q

S* = source/sink term in units of (M/Lt)

Equation 21 must be written for every transported species.

Initial and Boundary Conditions

Initial conditions

29. For continuity, the only initial conditions to be specified at time $t = t_o$ for all the nodes, i , for i = 1 to N are:

$$A(t = t_o, x) = A_i(x) \qquad (22)$$

or since there is a direct correspondence between A and h , it is permissible to use

$$h(t = t_o, x) = h_i(x) \qquad (23)$$

Either Equation 22 or 23 must be used, but not both.

30. For the momentum equation either

$$Q(t = t_o,x) = Q_i(x) \tag{24}$$

or

$$U(t = t_o,x) = U_i(x) \tag{25}$$

are permissible.

31. For the species transport equation, the following is used:

$$\alpha(t = t_o,x) = \alpha_i(x) \tag{26}$$

Boundary conditions for
continuity and momentum

32. For the upstream boundary at $x = 0$ and the downstream boundary at $x = L$, the following sets of boundary conditions may be used for the combined set of continuity and momentum equations. Please note that two first-order equations require only a total of two boundary conditions. Therefore, only one set may be used for each simulation.

$$h(t,x = 0) = h_u(t) \text{ and } h(t,x = L) = h_d(t) \tag{27}$$

$$Q(t,x = 0) = Q_u(t) \text{ and } Q(t,x = L) = Q_d(t) \tag{28}$$

$$h(t,x = 0) = h_u(t) \text{ and } Q(t,x = L) = Q_d(t) \tag{29}$$

$$Q(t,x = 0) = Q_u(t) \text{ and } h(t,x = L) = h_d(t) \tag{30}$$

$$h(t,x = 0) = h_u(t) \text{ and } f_1(Q,h) = f_{1d}(t) \tag{31}$$

$$Q(t,x = 0) = Q_u(t) \text{ and } f_2(Q,h) = f_{2d}(t) \tag{32}$$

where f_1 and f_2 are rating curves relating h and Q for the downstream, d , boundary conditions. It is permissible to properly substi te A and U for Q in these expressions.

33. It should be noted that the upstream boundary condition for a tributary can be selected by specifying either the elevation or the flow. At the downstream tributary boundary, i.e. the confluence with the main stem, only the elevation is allowed as a "boundary" condition to ensure that continuity is preserved at the junction.

Boundary conditions
for species transport

34. The boundary conditions for species transport are

$$\alpha(t,x = 0) = \alpha_u(t)$$

$$\alpha(t,x = L) = \alpha_d(t) \tag{33}$$

However, the condition $\alpha_d(t)$ is presently not specified as input data for riverine simulations. This must be included as input if the model is modified for inflow at the downstream end, such as occurs with estuarine boundary conditions.

Equation Summary

35. The following equations govern the unsteady, 1-D (longitudinal) hydrodynamics and transport:

Continuity

$$\frac{\partial(A + A_o)}{\partial t} + \frac{\partial Q}{\partial x} = q \tag{34}$$

Momentum

$$\frac{\partial Q}{\partial t} + \frac{\partial(UQ)}{\partial x} + gA\frac{\partial h}{\partial x} = gA\left(S_o - S_f - \frac{h_E}{\Delta x}\right) + qU_q \tag{35}$$

Constituent transport

$$\frac{\partial(\alpha A)}{\partial t} + \frac{\partial(Q\alpha)}{\partial x} = \frac{\partial}{\partial x}\left(DA\,\frac{\partial \alpha}{\partial x}\right) + \gamma q + S^{*} \tag{36}$$

The initial conditions are Equatic s 22 (or 23), 24 (or 25), and 26 or the continuity, momentum, and mass species equations rec ectively. Boundary conditions are one set selected from Equations 27 through 32 for the momentum and continuity equation and Equation 33 for the mass species equation.

PART III: THE NUMERICAL SOLUTION FOR FLOW AND ELEVATION

Rationale

36. As noted, flood and peaking hydropower waves and associated transport are marked by rapidly varying flows, elevations, and concentrations. As derived, the continuity and momentum equations, from which elevations and flows are calculated, are hyperbolic. The transport equation is dominated by advection; thus it has hyperbolic features. All of these equations, then, are very difficult to solve numerically because the smoothing, stabilizing effects of dispersion are eliminated or reduced. Inspection of the governing equations reveals that they are coupled, unsteady, and nonlinear, but because the contaminant concentrations do not affect the flow field, it is possible to uncouple the solution of the continuity and momentum equations from the solution of the species transport equation. Once solved, the complete time histories of flow and elevation can be stored and used as input information for the transport calculations. Because the governing equations are hyperbolic, the solution procedure for the flows and elevations is quite different from the transport solution; therefore, this chapter presents the solution procedure for the continuity and momentum equation.

Numerical Approximations

37. Three numerical procedures are useful for hyperbolic equations: the finite element method, the method of characteristics, and the finite difference implicit method. The method of characteristics is quite accurate, but can be difficult to program by anyone but a specialist. Reviews of this method are found in Liggett and Cunge (1975) and Abbott (1979). The implicit methods are simpler to program because they are much more direct numerical approximation techniques to partial derivatives. Implicit methods also possess favorable stability behavior even in applications with variable space and time steps. Explicit methods are totally unsatisfactory and are not considered.

38. There are many implicit procedures, but the method to be used here is the four-point implicit method first used by Preissmann (1961) with subsequent applications by, among others, Amein and Fang (1970) and Amein and

Chu (1975). This formulation is currently being used by Fread (1973, 1978) in the National Weather Service Dambreak Model (Fread 1978). The method is weighted implicit at each time level, is unconditionally stable for $0.5 < \theta \leq 1.0$, and permits relatively unequal space and time-steps. The scheme has second-order accuracy when $\theta = 0.5$ and first-order accuracy when $\theta = 1.0$. It is fully nonlinear but yet is a compact scheme requiring just two points at each time level for second-order spatial accuracy.

39. The river system is discretized (Figure 4) by a network of time and space nodes separated by time and space increments Δx_i, Δt_j. If β denotes the point about which the governing equation is to be discretized, then the values of the variables at the four points surrounding β are used to form the appropriate derivatives and weighted averages. For a general variable ω, then

$$\omega(\beta) \simeq \theta\left(\frac{\omega_i^{j+1} + \omega_{i+1}^{j+1}}{2}\right) + (1 - \theta)\left(\frac{\omega_i^{j} + \omega_{i+1}^{j}}{2}\right) \tag{37}$$

$$\frac{\partial \omega(\beta)}{\partial x} \simeq \theta\left(\frac{\omega_{i+1}^{j+1} - \omega_i^{j+1}}{\Delta x_i}\right) + (1 - \theta)\left(\frac{\omega_{i+1}^{j} - \omega_i^{j}}{\Delta x_i}\right) \tag{38}$$

$$\frac{\partial \omega(\beta)}{\partial t} \simeq \left(\frac{\omega_i^{j+1} + \omega_{i+1}^{j+1}}{2\Delta t_j}\right) - \left(\frac{\omega_i^{j} + \omega_{i+1}^{j}}{2\Delta t_j}\right) \tag{39}$$

Application to Governing Equations

The continuity equation

40. From Equation 34, the continuity equation is

$$\frac{\partial (A + A_o)}{\partial t} + \frac{\partial Q}{\partial x} - q = 0 \tag{40}$$

Using the definitions in Equations 37, 38, and 39, the discretized form of this equation is

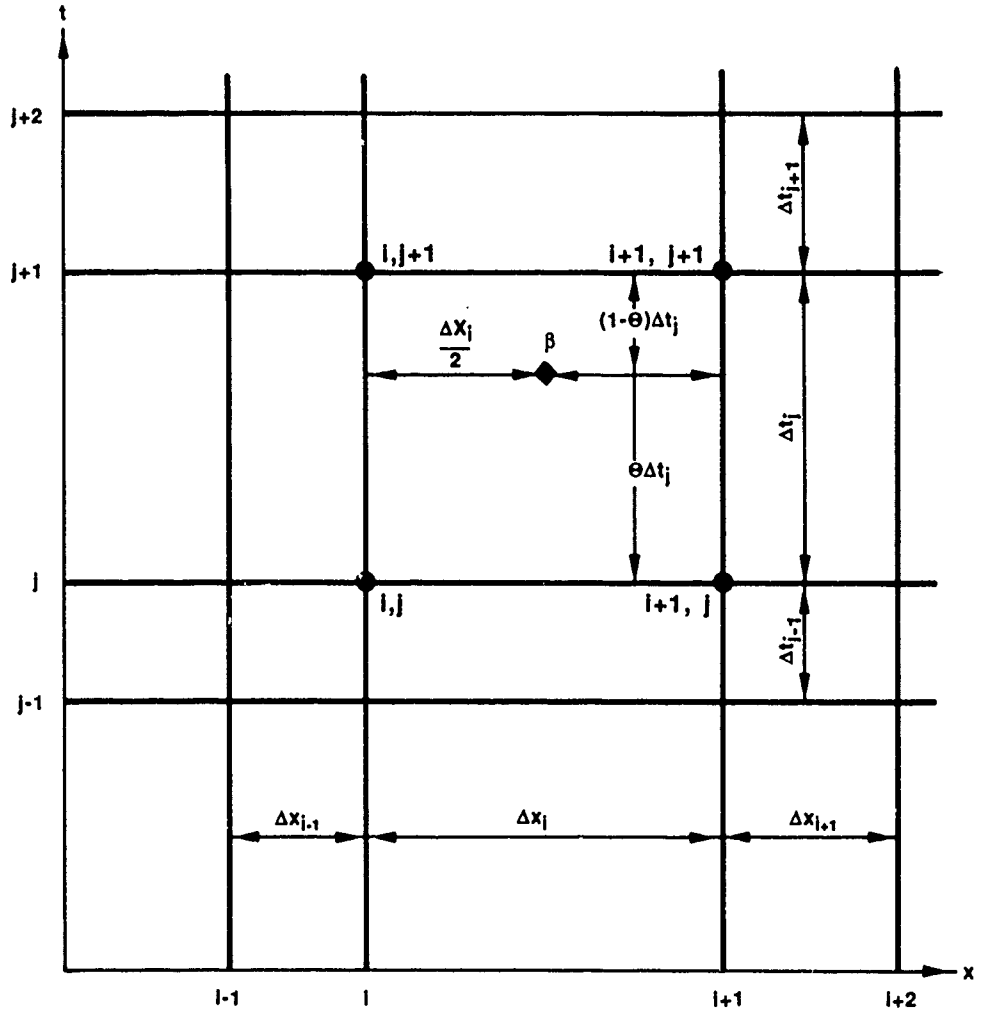


Figure 4. Numerical grid

$$\frac{1}{2\Delta t_j}\left[\left(A + A_o\right)_i^{j+1} + \left(A + A_o\right)_{i+1}^{j+1} - \left(A + A_o\right)_i^{j} - \left(A + A_o\right)_{i+1}^{j}\right]$$

$$+ \theta\left[\frac{1}{\Delta x_i}\left(Q_{i+1}^{j+1} - Q_i^{j+1}\right)\right] - \theta\,\frac{\left(q_{i+1}^{j+1} + q_i^{j+1}\right)}{2}$$

$$+ (1 - \theta)\left[\frac{1}{\Delta x_i}\left(Q_{i+1}^{j} - Q_i^{j}\right)\right] - (1 - \theta)\,\frac{\left(q_{i+1}^{j} + q_i^{j}\right)}{2}$$

$$= F_i\left(Q_{i+1}, A_{i+1}, Q_i, A_i\right) = 0 \qquad (41)$$

The momentum equation

41. From Equation 35 and using the definitions for S_E and S_f from Equations 20 and 11, the numerical discretization of the momentum equation is:

$$\frac{1}{2\Delta t_j}\left(Q_i^{j+1} + Q_{i+1}^{j+1} - Q_i^j - Q_{i+1}^j\right)$$

$$+ \theta\left\{\frac{1}{\Delta x_i}\left[\left(\frac{Q^2}{A}\right)_{i+1}^{j+1} - \left(\frac{Q^2}{A}\right)_i^{j+1}\right]\right\} + \theta g\left(\frac{A_{i+1}^{j+1} + A_i^{j+1}}{2}\right)\left[\frac{1}{\Delta x_i}\left(h_{i+1}^{j+1} - h_i^{j+1}\right)\right]$$

$$+ \frac{g\theta}{2(2.2)}\left[\left(\frac{n_{i+1}^2 \left|Q_{i+1}\right| Q_{i+1}}{A_{i+1} R_{i+1}^{4/3}}\right)^{j+1} + \left(\frac{n_i^2 \left|Q_i\right| Q_i}{A_i R_i^{4/3}}\right)^{j+1}\right]$$

$$- \theta g\left(\frac{A_{i+1}^{j+1} + A_i^{j+1}}{2}\right)\left(\frac{So_{i+1} + So_i}{2}\right)$$

$$+ \theta\left(\frac{A_{i+1}^{j+1} + A_i^{j+1}}{8}\right)\left\{\frac{K_E}{\Delta x_i}\left[\left(\frac{Q^2}{A^2}\right)_{i+1}^{j+1} + \left(\frac{Q^2}{A^2}\right)_i^{j+1}\right]\right\}$$

$$- \theta\left(\frac{q_{i+1}^{j+1} + q_i^{j+1}}{2}\right)\left(\frac{U_{q_{i+1}}^{j+1} + U_{q_i}^{j+1}}{2}\right)$$

$$+ (1 - \theta)\left\{\frac{1}{\Delta x_i}\left[\left(\frac{Q^2}{A}\right)_{i+1}^{j} - \left(\frac{Q^2}{A}\right)_i^{j}\right]\right\} + g(1 - \theta)\left(\frac{A_{i+1}^j + A_i^j}{2}\right)\left[\frac{1}{\Delta x_i}\left(h_{i+1}^j - h_i^j\right)\right]$$

$$+ \frac{g(1 - \theta)}{2(2.2)}\left[\left(\frac{n_{i+1}^2 \left|Q_{i+1}\right| Q_{i+1}}{A_{i+1} R_{i+1}^{4/3}}\right)^{j} + \left(\frac{n_i^2 \left|Q_i\right| Q_i}{A_i R_i^{4/3}}\right)^{j}\right]$$

$$- g(1 - \Theta) \left(\frac{A^j_{i+1} + A^j_i}{2} \right) \left(\frac{So_{i+1} + So_i}{2} \right)$$

$$+ (1 - \Theta) \left(\frac{A^j_{i+1} + A^j_i}{8} \right) \left\{ \frac{K_E}{\Delta x_i} \left[\left(\frac{Q^2}{A^2} \right)^j_{i+1} + \left(\frac{Q^2}{A^2} \right)^j_i \right] \right\}$$

$$- (1 - \Theta) \left(\frac{q^j_{i+1} + q^j_i}{2} \right) \left(\frac{U^j_{q_{i+1}} + U^j_{q_i}}{2} \right)$$

$$= G_i \left(Q_{i+1}, A_{i+1}, Q_i, A_i \right) = 0 \qquad (42)$$

The boundary conditions

42. The boundary conditions must also be "discretized." This is accomplished as follows. From the list of permissible pairs of boundary conditions, one set is selected, say Equation 27. Then the discretized form of the upstream boundary in terms of A becomes

$$F_o = A_1^{j+1} - A_u(t^{j+1}) = 0 \qquad (43)$$

while the downstream boundary condition at node N becomes

$$F_N = A_N^{j+1} - A_d(t^{j+1}) = 0 \qquad (44)$$

If flows or discharges are to be specified, then from Equation 28 the upstream boundary condition becomes

$$F_o = Q_1^{j+1} - Q_u(t^{j+1}) = 0 \qquad (45)$$

and the downstream condition is

$$F_N = Q_N^{j+1} - Q_d(t^{j+1}) = 0 \qquad (46)$$

Equation Assembly

43. Equations F_i (Equation 41) and G_i (Equation 42) are applied at every node i on both the main stem and any included tributaries. The system of algebraic equations results in 2N-2 equations for 2N unknowns. The two boundary conditions are sufficient to completely close the problem. If $G_o(Q_1,A_1)$ and $G_N(Q_N,A_N)$ are the boundary conditions written in the form of Equations 43-46, then the resulting system of 2N nonlinear equations is formally written as:

$$
\begin{aligned}
&G_0(Q_1,A_1) = 0 \\
&F_1(Q_2,A_2,Q_1,A_1) = 0 \\
&G_1(Q_2,A_2,Q_1,A_1) = 0 \\
&\cdots\cdots\cdots\cdots\cdots \\
&\cdots\cdots\cdots\cdots\cdots \\
&F_i(Q_{i+1},A_{i+1},Q_i,A_i) = 0 \\
&G_i(Q_{i+1},A_{i+1},Q_i,A_i) = 0 \\
&\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
&\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
&F_{N-1}(Q_N,A_N,Q_{N-1},A_{N-1}) = 0 \\
&G_{N-1}(Q_N,A_N,Q_{N-1},A_{N-1}) = 0 \\
&G_N(Q_N,A_N) = 0
\end{aligned}
\qquad (47)
$$

44. The general solution of these nonlinear equations can proceed in two ways. First the nonlinear terms may be linearized by using information from the j^{th} time-step. Time marching proceeds very easily by this procedure.

However, under the potentially rapid and sharp gradients being reproduced here, a full nonlinear solution by Newton-Raphson iteration procedures is recommended and implemented.

Newton-Raphson Solution for Flow and Elevation

45. The Newton-Raphson procedure is based upon estimating the residuals from each algebraic equation introduced by assuming initial answers for Q_i and A_i. The iterative reduction of the residuals by successive Taylor series refinement of the estimates of Q_i and A_i completes the procedure. The generalized Newton-Raphson procedure must be performed at each time-step. This method has been used with excellent success by the previously cited authors.

46. However, unlike these authors, this formulation employs a direct simultaneous solution at each iteration for all the flows and elevations in both main stem and tributaries. Previous methods employed an iterative method of estimating such variables. Therefore, for each Newton-Raphson iteration, another sequence of interior iterations was performed. The formulation herein requires only one five-band matrix solution at each Newton-Raphson iteration. The Newton-Raphson concept is presented herein, and details of the matrix packing and solution procedure are presented in the program operation portion of the manual.

47. The residuals, R_{1i}^k and R_{2i}^k, from the k^{th} estimate of Q_i and A_i are found for Equations F_i and G_i as

$$G_0\left(Q_1^k, A_1^k\right) = R_{2,0}^k$$

$$F_1\left(Q_2^k, A_2^k, Q_1^k, A_1^k\right) = R_{1,1}^k$$

$$G_1\left(Q_2^k, A_2^k, Q_1^k, A_1^k\right) = R_{2,1}^k$$

.....

.....

$$F_i\left(Q_{i+1}^k, A_{i+1}^k, Q_i^k, A_i^k\right) = R_{1,i}^k$$

$$G_i\left(Q_{i+1}^k, A_{i+1}^k, Q_i^k, A_i^k\right) = R_{2,i}^k$$

.....

.....

$$F_{N-1}\left(Q_N^k, A_N^k, Q_{N-1}^k, A_{N-1}^k\right) = R_{1,N-1}^k$$

$$G_{N-1}\left(Q_N^k, A_N^k, Q_{N-1}^k, A_{N-1}^k\right) = R_{2,N-1}^k$$

$$G_N\left(Q_N^k, A_N^k\right) = R_{2,N}^k \tag{48}$$

48. A generalized Taylors series for a function T that is a function of four dependent variables, S_1, S_2, S_3, and S_4 is written as

$$T^{k+1} = T^k + \frac{\partial T}{\partial S_1} dS_1 + \frac{\partial T}{\partial S_2} dS_2 + \frac{\partial T}{\partial S_3} dS_3 + \frac{\partial T}{\partial S_4} dS_4 \tag{49}$$

This general form is now used to relate the residuals to the gradients as follows. Keep in mind that we want to drive the residuals to zero; thus we want F_i and G_i to be zero. If T^{k+1} represents the exact solution for F_i or G_i, then T^{k+1} is zero. This means that the unknowns are dS_1, dS_2, dS_3, and dS_4. Let T^k be the R_{ji}^k residual. Further, assume that S_1, S_2, S_3, and S_4 become the values Q_{i+1}, A_{i+1}, Q_i, A_i. In the Newton-Raphson iteration, the gradients are known from the k^{th} estimate of Q_i and A_i, and the relationship between the gradients and the residuals becomes

$$\frac{\partial G_0}{\partial A_1} dA_1 + \frac{\partial G_0}{\partial Q_1} dQ_1 = R_{2,0}^k$$

$$\frac{\partial F_1}{\partial Q_2} dQ_2 + \frac{\partial F_1}{\partial A_2} dA_2 + \frac{\partial F_1}{\partial Q_1} dQ_1 + \frac{\partial F_1}{\partial A_1} dA_1 = R^k_{1,1}$$

$$\frac{\partial G_1}{\partial Q_2} dQ_2 + \frac{\partial G_1}{\partial A_2} dA_2 + \frac{\partial G_1}{\partial Q_1} dQ_1 + \frac{\partial G_1}{\partial A_1} dA_1 = R^k_{2,1}$$

.....

.....

$$\frac{\partial F_i}{\partial Q_{i+1}} dQ_{i+1} + \frac{\partial F_i}{\partial A_{i+1}} dA_{i+1} + \frac{\partial F_i}{\partial Q_i} dQ_i + \frac{\partial F_i}{\partial A_i} dA_i = R^k_{1,i}$$

$$\frac{\partial G_i}{\partial Q_{i+1}} dQ_{i+1} + \frac{\partial G_i}{\partial A_{i+1}} dA_{i+1} + \frac{\partial G_i}{\partial Q_i} dQ_i + \frac{\partial G_i}{\partial A_i} dA_i = R^k_{2,i}$$

.....

$$\frac{\partial F_{N-1}}{\partial Q_N} dQ_N + \frac{\partial F_{N-1}}{\partial A_N} dA_N + \frac{\partial F_{N-1}}{\partial Q_{N-1}} dQ_{N-1} + \frac{\partial F_{N-1}}{\partial A_{N-1}} dA_{N-1} = R^k_{1,N-1}$$

$$\frac{\partial G_{N-1}}{\partial Q_N} dQ_N + \frac{\partial G_{N-1}}{\partial A_N} dA_N + \frac{\partial G_{N-1}}{\partial Q_{N-1}} dQ_{N-1} + \frac{\partial G_{N-1}}{\partial A_{N-1}} dA_{N-1} = R^k_{2,N-1}$$

$$\frac{\partial G_N}{\partial Q_N} dQ_N + \frac{\partial G_N}{\partial A_N} dA_N = R^k_{2,N} \qquad (50)$$

where the minus sign (from moving T^k to the other side of the equation) has been included in the residuals; thus the residuals are the negative of Equations 41 and 42. Note that all derivatives are taken with respect to Q^{j+1}_{i+1}, A^{j+1}_{i+1}, Q^{j+1}_i, A^{j+1}_i, etc. Additionally, for all Q_i and A_i evaluated at time $j+1$

$$dQ_1 = Q_1^{k+1} - Q_1^k$$

$$dA_1 = A_1^{k+1} - A_1^k$$

.....
.....

$$dQ_i = Q_i^{k+1} - Q_i^k$$

$$dA_i = A_i^{k+1} - A_i^k$$

.....
.....

$$dQ_N = Q_N^{k+1} - Q_N^k$$

$$dA_N = A_N^{k+1} - A_N^k \tag{51}$$

49. The derivatives of F_i and G_i with respect to A_{i+1}^{j+1}, Q_{i+1}^{j+1}, A_i^{j+1}, and Q_i^{j+1} are found from Equations 41 and 42 as follows:

$$\frac{\partial F_i}{\partial A_i^{j+1}} = \frac{1}{2\Delta t_j} \tag{52}$$

$$\frac{\partial F_i}{\partial Q_i^{j+1}} = -\frac{\theta}{\Delta x_i} \tag{53}$$

$$\frac{\partial F_i}{\partial A_{i+1}^{j+1}} = \frac{1}{2\Delta t_j} \tag{54}$$

$$\frac{\partial F_i}{\partial Q_{i+1}^{j+1}} = \frac{\theta}{\Delta x_i} \tag{55}$$

$$\frac{\partial G_i}{\partial Q_i^{j+1}} = \frac{1}{2\Delta t_j} + \theta\left[\frac{-2}{\Delta x_i}\frac{Q_i^{j+1}}{A_i^{j+1}} + \frac{g}{2(2.2)}\frac{n_i^2\left|Q_i^{j+1}\right|}{A_i^{j+1}\left(R_i^{j+1}\right)^{4/3}} + 2\left(\frac{A_{i+1}^{j+1} + A_i^{j+1}}{8\Delta x_i}\right)K_E\frac{Q_i^{j+1}}{\left(A_i^{j+1}\right)^2}\right] \tag{56}$$

$$\frac{\partial G_i}{\partial Q_{i+1}^{j+1}} = \frac{1}{2\Delta t_j} + \theta\left[\frac{2}{\Delta x_i}\frac{Q_{i+1}^{j+1}}{A_{i+1}^{j+1}} + \frac{g}{2(2.2)}\frac{n_{i+1}^2\left|Q_{i+1}^{j+1}\right|}{A_{i+1}^{j+1}\left(R_{i+1}^{j+1}\right)^{4/3}} + 2\left(\frac{A_{i+1}^{j+1} + A_i^{j+1}}{8\Delta x_i}\right)K_E\frac{Q_{i+1}^{j+1}}{\left(A_{i+1}^{j+1}\right)^2}\right] \tag{57}$$

$$\frac{\partial G_i}{\partial A_{i+1}^{j+1}} = \theta\left\{\frac{-1}{\Delta x_i}\left(\frac{Q^2}{A^2}\right)_{i+1}^{j+1} + \frac{g}{2\Delta x_i}\left[\left(h_{i+1}^{j+1} - h_i^{j+1}\right) + \frac{\left(A_{i+1}^{j+1} + A_i^{j+1}\right)}{B_{i+1}^{j+1}}\right]\right.$$

$$+ \frac{gn_{i+1}^2}{6(2.2)}\frac{\left|Q_{i+1}^{j+1}\right|Q_{i+1}^{j+1}}{A_{i+1}^{j+1}\left(R_{i+1}^{j+1}\right)^{4/3}}\left[\frac{-7}{A_{i+1}^{j+1}} + \frac{4\frac{dB}{dh}\Big|_{i+1}^{j+1}}{\left(B_{i+1}^{j+1}\right)^2} + \frac{6\frac{\partial n}{\partial h}\Big|_{i+1}}{n_{i+1}B_{i+1}^{j+1}}\right]$$

$$\left. - \frac{g}{2}\left(\frac{S_{o_{i+1}} + S_{o_i}}{2}\right) + \frac{k_E}{8\Delta x_i}\left[\left(\frac{Q^2}{A^2}\right)_i^{j+1} - \left(\frac{Q^2}{A^2}\right)_{i+1}^{j+1}\left(1 + \frac{2A_i^{j+1}}{A_{i+1}^{j+1}}\right)\right]\right\} \tag{58}$$

$$\frac{\partial G_i}{\partial A^{j+1}} = \theta\left\{\frac{1}{\Delta x_i}\left(\frac{Q^2}{A^2}\right)_i^{j+1} + \frac{g}{2\Delta x_i}\left[\left(h_{i+1}^{j+1} - h_i^{j+1}\right) - \frac{\left(A_{i+1}^{j+1} + A_i^{j+1}\right)}{B_i^{j+1}}\right]\right.$$

$$+ \frac{gn_i^2}{6(2.2)} \frac{\left|Q_i^{j+1}\right| Q_i^{j+1}}{A_i^{j+1}\left(R_i^{j+1}\right)^{4/3}}\left[\frac{-7.}{A_i^{j+1}} + \frac{4\left.\frac{dB}{dh}\right|_i^{j+1}}{\left(B_i^{j+1}\right)^2} + \frac{6\left.\frac{\partial n}{\partial h}\right|_i}{n_i B_i^{j+1}}\right]$$

$$\left. - \frac{g}{2}\left(\frac{S_{o_{i+1}} + S_{o_i}}{2}\right) + \frac{K_E}{8\Delta x_i}\left[\left(\frac{Q^2}{A^2}\right)_{i+1}^{j+1} - \left(\frac{Q^2}{A^2}\right)_i^{j+1}\left(1 + \frac{2A_{i+1}^{j+1}}{A_i^{j+1}}\right)\right]\right\} \tag{59}$$

50. The following subsidiary manipulations have been used. First, area and depth gradients are related by the equation $\partial A/\partial x = B\ \partial h/\partial x$ and second, the hydraulic radius is approximately equal to the hydraulic depth or $R \simeq A/B$. The explicit relationship between A and h is permitted if the general empirical form for most channel cross sections is used, i.e.

$$A = a_o h + a_1 h^{a_2} \tag{60}$$

By appropriate selection of the coefficients, most regularly formed channel cross sections can be modeled by this function. A library of such shapes was created and will be described in detail in Part VII of this report. It is also noted that Equations 52-59 have been multiplied by $2\Delta t_j$ in the code.

Calculation Procedure

51. The nonlinear solution procedure is used to calculate the new flows, Q_i^{j+1}, and areas, A_i^{j+1}, as follows:

a. Assume that Q_i^j and A_i^j are known either from initial conditions or from the completion of the previous Newton Raphson solution.

b. By insertion of Q_i^j and A_i^j into the equations for F and G, form the $k = 1$ residuals R_{ji}^i, by assuming that $k = 1$

estimate for $Q_i^{j+1,k}$ and $A_i^{j+1,k}$ is the j^{th} time-step value for Q and A or the initial condition (Step 1). For $k > 1$, the previous k^{th} estimate of Q_i^{j+1} and A_i^{j+1} are used.

c. After forming the residuals, the gradients are formed from Equations 52-59, again by using the $k = 1$ estimates for Q and A. For subsequent iterations, the previous k^{th} estimates are used.

$$
\begin{bmatrix}
\frac{\partial G_0}{\partial A_1} & \frac{\partial G_0}{\partial Q_1} & & & & & & & & & & \\
\frac{\partial F_1}{\partial A_1} & \frac{\partial F_1}{\partial Q_1} & \frac{\partial F_1}{\partial A_2} & \frac{\partial F_1}{\partial Q_2} & & & & & & & & \\
\frac{\partial G_1}{\partial A_1} & \frac{\partial G_1}{\partial Q_1} & \frac{\partial G_1}{\partial A_2} & \frac{\partial G_1}{\partial Q_2} & & & & & & & & \\
 & \cdots\cdots\cdots\cdots & & & & & & & & & & \\
 & \cdots\cdots\cdots\cdots & & & & & & & & & & \\
 & & \frac{\partial F_i}{\partial A_i} & \frac{\partial F_i}{\partial Q_i} & \frac{\partial F_i}{\partial A_{i+1}} & \frac{\partial F_i}{\partial Q_{i+1}} & & & & & & \\
 & & \frac{\partial G_i}{\partial A_i} & \frac{\partial G_i}{\partial Q_i} & \frac{\partial G_i}{\partial A_{i+1}} & \frac{\partial G_i}{\partial Q_{i+1}} & & & & & & \\
 & & \cdots\cdots\cdots\cdots & & & & & & & & & \\
 & & \cdots\cdots\cdots\cdots & & & & & & & & & \\
 & & & & & & \frac{\partial F_{N-1}}{\partial A_{N-1}} & \frac{\partial F_{N-1}}{\partial Q_{N-1}} & \frac{\partial F_{N-1}}{\partial A_N} & \frac{\partial F_{N-1}}{\partial Q_N} & & \\
 & & & & & & \frac{\partial G_{N-1}}{\partial A_{N-1}} & \frac{\partial G_{N-1}}{\partial Q_{N-1}} & \frac{\partial G_{N-1}}{\partial A_N} & \frac{\partial G_{N-1}}{\partial Q_N} & & \\
 & & & & & & & & & & \frac{\partial G_N}{\partial A_N} & \frac{\partial G_N}{\partial Q_N}
\end{bmatrix}
\begin{bmatrix} dA_1^k \\ dQ_1^k \\ dA_2^k \\ dQ_2^k \\ \cdots \\ \cdots \\ dA_i^k \\ dQ_i^k \\ dA_{i+1}^k \\ dQ_{i+1}^k \\ \cdots \\ \cdots \\ dA_{N-1}^k \\ dQ_{N-1}^k \\ dA_N^k \\ dQ_N^k \end{bmatrix}
=
\begin{bmatrix} R_{2,0}^k \\ R_{1,1}^k \\ R_{2,1}^k \\ R_{1,2}^k \\ \cdots \\ \cdots \\ R_{1,i}^k \\ R_{2,i}^k \\ R_{1,i+1}^k \\ R_{2,i+1}^k \\ \cdots \\ \cdots \\ R_{1,N-1}^k \\ R_{2,N-1}^k \\ R_{2,N}^k \end{bmatrix}
\qquad (61)
$$

d. From Equation 50 and Steps 1 through 3, a coefficient matrix is formed and assembled as in Equation 61; a system of linear simultaneous equations $[M]^k\{D\}^k = \{R\}^k$ is solved for $\{D\}^k$.

e. When solved for, the departure vector $\{D\}^k$ is added to the old estimates of Q and A, to give a new estimate for Q and A, i.e.

$$Q_i^{j+1,k+1} = Q_i^{j+1,k} + dQ_i^k \tag{62}$$

and

$$A_i^{j+1,k+1} = A_i^{j+1,k} + dA_i^k \tag{63}$$

f. A check is made to see how close $Q_i^{j+1,k+1}$ and $A_i^{j+1,k+1}$ are to $Q_i^{j+1,k}$ and $A_i^{j+1,k}$ and if the largest difference is less than some specified tolerance, the iteration stops and the new values of A and Q for the $j+1$ time-steps are at hand. If the tolerance is exceeded, return to Step 2 and using $Q_i^{j+1,k+1}$ and $A_i^{j+1,k+1}$, repeat Steps 2 through 6.

52. The programming of this routine is discussed in the program structure section, Part VII.

PART IV: THE NUMERICAL SOLUTION FOR CONSTITUENT TRANSPORT

Rationale

53. By inspection of the governing equations for flow and elevation (Equations 34 and 35) and constituent concentration (Equation 36), it is apparent that the flow field is not affected by constituent concentration. A complete prediction of Q and A can be made without one's solving for the constituents. This is convenient since the constituent equation can be solved separately providing economy. Since any number of species could conceivably be solved for, the numerical solution must be as quick as possible. This suggests the use of explicit time-marching procedures. However, simple explicit (and also simple implicit methods) time marching for the advection problem is a very severe test for which, unfortunately, the simple methods fail to provide the desired accuracy. It should be noted that pure advection is the single most difficult test for a numerical method since the initial concentration distribution imposed on the problem must be numerically advected or transported without loss of mass, shape, and peak value, or distortion of the statistics of the distribution including mean, variance, skew, and kurtosis.

54. A powerful and accurate explicit method based upon compact, but fourth-order accurate, numerical expressions is used to solve Equation 36 for advection. An implicit fractional step method is subsequently used for the dispersion term.

The Governing Equation

55. The general form of the equation to be solved is, for a mass concentration α:

$$\frac{\partial(A\alpha)}{\partial t} + \frac{\partial(UA\alpha)}{\partial x} = \frac{\partial}{\partial x}\left(DA\,\frac{\partial \alpha}{\partial x}\right) + q\gamma + S^* \tag{64}$$

where

D = dispersion coefficient

γ = concentration of the runoff input to the channel by distributed flow q

S^* = source/sink term which accounts for changes in α due solely to biological and chemical reasons

The variable S^* is decomposed into a source/sink term that is a function of the present concentration of α and a function that is not; therefore

$$\frac{\partial(A\ \alpha)}{\partial t} + \frac{\partial(Q\ \alpha)}{\partial x} = \frac{\partial}{\partial x}\left(DA\ \frac{\partial \alpha}{\partial x}\right) + q\gamma + AC_1\alpha + AC_2 \tag{65}$$

where C_1 has units of $(1/t)$ and C_2 has units of (M/L^3t). This is the form of the equation to be solved, and it requires the initial and boundary conditions as stipulated in Equations 26 and 33, respectively.

56. The governing equation after chain rule differentiation is reassembled in the form

$$\frac{\partial \alpha}{\partial t} + U\ \frac{\partial \alpha}{\partial x} + \frac{\alpha\ \partial U}{\partial x} - \left(\frac{D}{A}\ \frac{\partial A}{\partial x} + \frac{\partial D}{\partial x}\right)\frac{\partial \alpha}{\partial x} = D\ \frac{\partial^2 \alpha}{\partial x^2} - \phi_1\alpha + \phi_2 \tag{66}$$

where

$$\phi_1 = \left(\frac{1}{A}\ \frac{\partial A}{\partial t} + \frac{U}{A}\ \frac{\partial A}{\partial x} - C_1\right) \tag{67}$$

and

$$\phi_2 = \left(C_2 + \frac{q\gamma}{A}\right) \tag{68}$$

Equation 66 can be rewritten

$$\frac{\partial \alpha}{\partial t} + \bar{u}\ \frac{\partial \alpha}{\partial x} = D\ \frac{\partial^2 \alpha}{\partial x^2} - \phi_1\alpha + \phi_2 - \alpha\ \frac{\partial U}{\partial x} \tag{69}$$

where

$$\bar{u} = U - \left(\frac{\partial D}{\partial x} + \frac{D}{A}\ \frac{\partial A}{\partial x}\right) = U - DDA \tag{70}$$

and

$$DDA = \frac{\partial D}{\partial x} + \frac{D}{A}\left(\frac{\partial A}{\partial x}\right) \tag{71}$$

From continuity (Equation 34)

$$\frac{\partial A}{\partial t} + A \frac{\partial U}{\partial x} + U \frac{\partial A}{\partial x} = q \tag{72}$$

Using Equation 72, the second and fourth terms on the right side of Equation 69 can be combined as

$$-\alpha\left(\phi_1 + \frac{\partial U}{\partial x}\right) = -\alpha\left(\frac{q}{A} - C_1\right) \tag{73}$$

and ϕ_1 is redefined as

$$\phi_1 = \frac{q}{A} - C_1 \tag{74}$$

Thus Equation 69 becomes

$$\frac{\partial \alpha}{\partial t} + \bar{u} \frac{\partial \alpha}{\partial x} = D \frac{\partial^2 \alpha}{\partial x^2} - \phi_1 \alpha + \phi_2 \tag{75}$$

57. The ϕ_1 and ϕ_2 terms of Equation 75 are written in expanded form so that they can be followed more easily in the code; thus

$$\frac{\partial \alpha}{\partial t} + \bar{u} \frac{\partial \alpha}{\partial x} = D \frac{\partial^2 \alpha}{\partial x^2} + \frac{q}{A} (\gamma - \alpha) - K_s \alpha + \text{SINKS} \tag{76}$$

where

$K_s = -C_1$ = biochemical uptake or decay rates (+) and growth rates (-)

SINKS = C_2 = biochemical sources (+) and sinks (-)

58. The left side of Equation 76 is solved for the new time level value of α (α_{i+1}^{j+1}) with the fourth-order explicit scheme. Using this new value, α_{i+1}^{j+1} is incremented due to the effects of the second, third, and fourth terms on the right side of Equation 76. The SINKS and K_s values are interpolated values between nodes (see Equation 90). Finally, the solution is completed by adding the effect of diffusion (first term on right side of Equation 76), which is computed implicitly.

Fourth-Order Explicit Scheme

59. The compact, fourth-order accurate scheme presented by Holly and Preissmann (1978) is a satisfactory scheme for 1-D advection calculations and is used for all mass transport calculations.

Polynomial assumption

60. It is assumed that the variation of any quantity between two adjacent spatial nodes is not linear but is depicted by a cubic polynomial such that for a variable Y

$$Y(\xi) = A\xi^3 + B\xi^2 + D\xi + E \tag{77}$$

where

$$\xi = \frac{u^*\tau}{x_{i+1} - x_i} \tag{78}$$

where

u^* = average characteristic velocity

τ = time-step $t_{i+1} - t_i$

The coefficients for the polynomial are evaluated from the conditions that

$$Y(1) = \alpha_i^j \; ; \; Y(0) = \alpha_{i+1}^i \; ; \; \dot{Y}(1) = \alpha x_i^j \; ; \; \dot{Y}(0) = \alpha x_{i+1}^j \tag{79}$$

$$\alpha x = \frac{\partial \alpha}{\partial x}$$

$$\dot{Y}(\xi) = \left.\frac{dY}{dx}\right|_{\xi}$$

After some algebra

$$Y(\xi) = A_1\, \alpha_i^j + A_2\, \alpha_{i+1}^j + A_3\, \alpha x_i^j + A_4\, \alpha x_{i+1}^j \tag{80}$$

where

$$A_1 = \xi^2(3 - 2\xi) \tag{81}$$

$$A_2 = 1 - A_1 \tag{82}$$

$$A_3 = \xi^2(1 - \xi)(x_{i+1} - x_i) \tag{83}$$

$$A_4 = -\xi(1 - \xi)^2(x_{i+1} - x_i) \tag{84}$$

Note that since the polynomial is parameterized by first derivatives, an equation for the first derivatives is also necessary. A polynomial for the first derivatives is formed from

$$\dot{Y}(\xi) = b_1\alpha_i^j + b_2\alpha_{i+1}^j + b_3\alpha x_i^j + b_4\alpha x_{i+1}^j \tag{85}$$

where

$$b_1 = 6\xi(\xi - 1)\left(x_{i+1} - x_i\right)^{-1} \tag{86}$$

$$b_2 = -b_1 \tag{87}$$

$$b_3 = \xi(3\xi - 2) \tag{88}$$

$$b_4 = (\xi - 1)(3\xi - 1) \tag{89}$$

Both $Y(\xi)$ and $\dot{Y}(\xi)$ will be used to determine α_{i+1}^{j+1} and αx_{i+1}^{j+1}, respectively, resulting from pure advective transport (the left side of Equation 76).

Solution procedure for α_{i+1}^{j+1}

61. Any variable or coefficient can be interpolated to obtain the average value between nodes as

$$K^* = \frac{\left[K_{i+1}^{j+1} + K_{i+1}^j\ (1 - \xi) + K_i^j\xi\right]}{2} \tag{90}$$

where

$$\xi = \frac{u^*\tau}{x_{i+1} - x_i}$$

and u^* is the average characteristic velocity between nodes. Likewise, u^* can be found by placing a linear interpolation between $\bar{u}_i$ and $\bar{u}_{i+1}$, based on u^*, or from Equation 90

$$u^* = \frac{\left[\bar{u}^{j+1}_{i+1} + \bar{u}^{j}_{i+1} - \left(\frac{u^*\tau}{x_{i+1} - x_i}\right)\left(\bar{u}^{j}_{i+1} - \bar{u}^{j}_{i}\right)\right]}{2} \tag{91}$$

solving for u^* and substituting Equation 70 for $\bar{u}$

$$u^* = \frac{U^{j+1}_{i+1} + U^{j}_{i+1} - DDA^{j+1}_{i+1} - DDA^{j}_{i+1}}{2 + \frac{\tau}{x_{i+1} - x_i}\left(U^{j}_{i+1} - U^{j}_{i}\right)} \tag{92}$$

The term $\frac{\partial DDA}{\partial x}$ becomes zero since D and A are allowed only linear variations between nodes.

62. The value of α^{j+1}_{i+1} due to advection, α^{**}_{i+1}, can now be determined from

$$\alpha^{**}_{i+1} = Y(\xi) \tag{93}$$

where $Y(\xi)$ is evaluated by Equations 80-84. The values for αx^{j}_{i} and αx^{j}_{i+1} in Equation 80 must be determined from

$$\alpha x^{j}_{i+1} = \dot{Y}(\xi) \tag{94}$$

or Equations 85-89. The decay, sources/sinks, and lateral inflow terms are next added to α^{**}_{i+1} such that

$$\alpha^{j+1}_{i+1} = \alpha^{**}_{i+1}\,(1 - \tau k_s) + \tau\left[SINKS + \frac{q}{A}\left(\gamma - \alpha^{**}_{i+1}\right)\right] \tag{95}$$

Solution procedure for αx_{i+1}^{j+1}

63. A spatial derivative of the transport equation must be developed to update αx_{i+1}^{j+1} . This development is done by taking the derivative of Equation 76 with respect to x ,

$$\frac{\partial \alpha'}{\partial t} + \bar{\bar{u}} \frac{\partial \alpha'}{\partial x} = D \frac{\partial^2 \alpha'}{\partial x^2} - \bar{u}' \alpha' + \left(\frac{q}{A}\right)' (\gamma - \alpha) - \frac{q}{A} \alpha' - k_s \alpha' - k_s \alpha' + SINKS' \quad (96)$$

where

$$\bar{\bar{u}} = \bar{u} - D'$$

and the prime denotes $\frac{\partial}{\partial x}$; thus $\alpha' = \frac{\partial \alpha}{\partial x} = \alpha x \equiv \alpha_x$. Now $\bar{u} = U'$ since D and A are allowed only linear variations between nodes. If D' is small and D at a node is constant or changes slowly over time, the characteristic velocity for the spatial gradient is approximated by $u^{**} = u^{*} - D'$. With the value u^{**} , ξ^{*} can be determined from

$$\xi^{*} = \frac{\tau u^{**}}{x_{i+1} - x_i} \quad (97)$$

64. Now $\dot{Y}(\xi^{*})$ can be evaluated through Equations 85-89. This evaluation yields $\alpha x_{i+1}^{**} = \dot{Y}(\xi^{*})$, which is the solution to the left side of Equation 96. All but the first term on the right side of Equation 96 are added to α_{i+1}^{**} such that

$$\alpha x_{i+1}^{j+1} = \alpha x_{i+1}^{**} \left[1 - \frac{\tau\left(U_{i+1}^{j} - U_i^{j}\right)}{x_{i+1} - x_i} - \tau\left(K_s + \frac{q}{A}\right)\right]$$

$$+ \tau \left[\left(\frac{q}{A}\right)' (\gamma - \alpha) - \alpha K_s' + SINKS'\right] \quad (98)$$

The final update for αx_{i+1}^{j+1} is completed when the implicit solution due to the first term on the right side of Equation 96 (diffusion) is added.

65. The derivatives for coefficients used in Equations 96-98 are found by

$$\frac{\partial K^j}{\partial x} \simeq \frac{K^j_{i+1} - K^j_i}{x_{i+1} - x_i} \tag{99}$$

and

$$\frac{\partial^2 k^j}{\partial x^2} \simeq \frac{\left(\left.\frac{\partial k^j}{\partial x}\right|_{i+1} - \left.\frac{\partial k^j}{\partial x}\right|_{i} \right)}{x_{i+1} - x_i} \tag{100}$$

The steps outlined in this section are repeated for each spatial node before moving on to the next time line.

Implicit Diffusion

66. The constituent transport (Equation 76) and the spatial derivative transport (Equation 96) equations are now ready to have the effects of diffusion added by

$$\alpha_i^{j+1} = \alpha_i^j + \tau D \frac{\partial^2 \alpha}{\partial x^2}$$

$$\alpha x_i^{j+1} = \alpha x_i^j + \tau D \frac{\partial^2 \alpha'}{\partial x^2} \tag{101}$$

where now the j time-level is actually at the new time-level following the advection and kinetic reactions but just prior to diffusion. The diffusion terms are approximated by difference equations, approximately centered in space and time. That is, new time information (j+1) is weighted by a factor $\theta = 0.55$ to enhance stability. The second derivative is replaced by the difference operator, D_{xx} ; thus

$$\frac{\partial^2 \alpha}{\partial x^2} = \theta D_{xx}(\alpha^{j+1}) + (1 - \theta)\, D_{xx}(\alpha^j) \qquad (102)$$

67. In space, centering would be exact for a regular grid. However, with an irregular grid, a quadratic interpolation is used by taking a Taylor series such that

$$D_{xx}(\alpha) = 2\left[\frac{\alpha_{i+1} - \alpha_i}{\Delta x_i(\Delta x_{i-1} + \Delta x_i)} + \frac{\alpha_{i-1} - \alpha_i}{\Delta x_{i-1}(\Delta x_{i-1} + \Delta x_i)}\right] \qquad (103)$$

where

$$\Delta x_i = x_{i+1} - x_i$$

Applying Equation 103 to the difference operator (Equation 102) which is subsequently applied to Equation 101 results in Equation 104 for α_i^{j+1},

$$\alpha_i^{j+1} = \alpha_i^j + 2\tau D\theta\left[\frac{\alpha_{i+1}^{j+1} - \alpha_i^{j+1}}{\Delta x_i(\Delta x_{i-1} + \Delta x_i)} + \frac{\alpha_{i-1}^{j+1} - \alpha_i^{j+1}}{\Delta x_{i-1}(\Delta x_{i-1} + \Delta x_i)}\right]$$

$$+ 2\tau D(1 - \theta)\left[\frac{\alpha_{i+1}^{j} - \alpha_i^{j}}{\Delta x_i(\Delta x_{i-1} + \Delta x_i)} + \frac{\alpha_{i-1}^{j} - \alpha_i^{j}}{\Delta x_{i-1}(\Delta x_{i-1} + \Delta x_i)}\right] \qquad (104)$$

An equation similar to Equation 104 is developed for αx_i^{j+1}.

68. When Equation 104 is applied to every node for i going from 2 to N-1, N-2 equations for N unknowns are formed. The system is completed by the upstream and downstream boundary conditions. The upstream boundary condition is given explicitly in the data (or inferred for the spatial derivative of diffusion). The downstream boundary condition is simply that the last node is not affected by diffusion.

69. The system of equations is assembled in tridiagonal form with all new values (j+1) on the left side and all old (j) values on the right. The implicit solution is accomplished with the Thomas Algorithm (subroutine TRIDAG). Subroutine TRIDAG is also used to calculate a cubic spline through the initial data.

Stability Requirements

70. The solution of the constituent transport equation has a Courant number restriction for numerical stability. This is due to the explicit solution scheme for advection. The Courant number, defined as

$$CN = \frac{U\tau}{\Delta x} \tag{105}$$

must be less than 1.0 to preserve stability. This requirement is not considered too restrictive, although it must be kept in mind during an application.

71. This version of the RIV1Q model is coded to expect flows in the downstream direction only. Erroneous numerical solutions can be experienced if flows in the upstream direction are experienced. Upstream flows do not usually occur in nontidal streams. However, highly unsteady flows may reflect off downstream control structures resulting temporarily in upstream "reverse flows." RIV1H can yield reverse flows that are realistic, but reverse flows that persist very long will yield unrealistic results from RIV1Q.

72. In a model application, it was possible to damp out reverse flows that were reflected off a dam downstream of a peaking hydropower dam by providing a minimum flow during nongeneration periods. This minimum flow was equivalent to the dam leakage. RIV1Q is being modified to allow reversing flows.

PART V: THE EFFECTS OF TURBULENCE AND TEMPERATURE UPON THE ASSIMILATIVE CAPACITY

Turbulence

73. Turbulence in streams affects both the rate of gas exchange with the atmosphere and the rate of pollutant removal. In both cases, the effects are achieved via the compression and disruption of the boundary layer at either the top or the bottom of the stream. While boundary layer theory suffices to account for virtually the whole of gas exchange, it must be borne in mind that pollutant removal occurs via three separate mechanisms: (a) biodegradation of suspended and dissolved matter by plankton; (b) sedimentation of settleable matter into the benthos, where it may be degraded; and (c) diffusion of soluble and colloidal matter through the boundary layer into the benthos, where pollutants react either biologically or physicochemically. Turbulence substantially affects only the last two removal mechanisms. The present model represents all three mechanisms by a single decay term. This limitation can be improved upon later if sediment transport and sediment/water interactions are included.

Rate of pollutant removal

74. Consider first the flux of matter into the benthos (Figure 5). A simplified version of Novotny's (1969) analysis is presented.

75. It is assumed that matter transits the benthal boundary layer (or viscous sublayer) by molecular diffusion and the time scales are long enough that the composition of the boundary layer at any given river station is constant. The concentration of the substance of interest is C (mass per volume) at the top of the boundary layer and zero, or at least much less than C, at the bottom. Under these conditions, Fick's law of diffusion can be written in the vertical direction as:

$$\text{Flux density}\left(\frac{\text{mass}}{\text{area} \cdot \text{time}}\right) = \frac{D_L C}{h_b} \tag{106}$$

where

D_L = molecular diffusivity of the substance, area/time

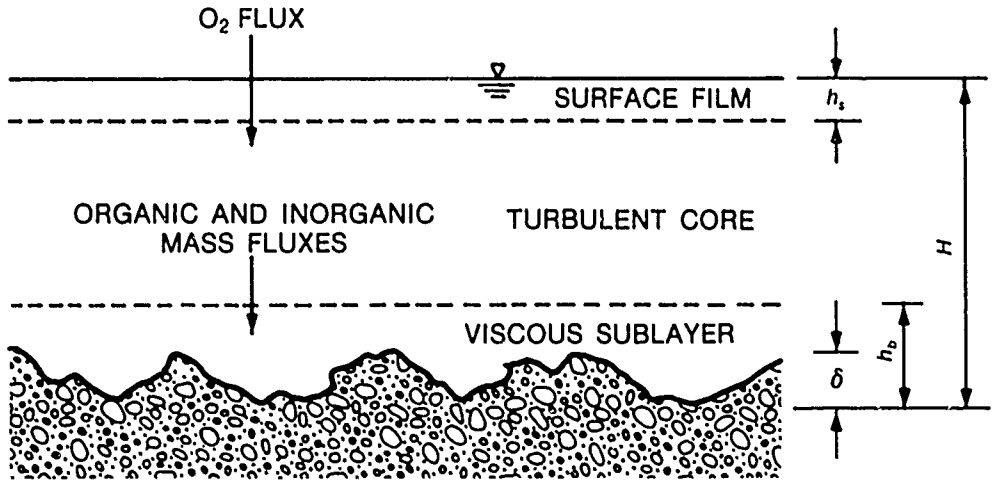


Figure 5. Schematic longitudinal section of a river

C = concentration of the substance in the main body of flow at the given river station, mass/volume

h_b = thickness of the benthal boundary layer, length

The problem here is to evaluate the thickness of the benthal boundary layer. Dimensional analysis shows that the ratio of the boundary layer thickness to the height of a characteristic roughness element should be a function of the boundary layer Reynold's number (Yalin 1977):

$$\frac{h_b}{\delta} = f\left(\frac{v_* \delta}{\nu}\right) \tag{107}$$

where

v_* = shear velocity; defined to be equal to the square root of the ratio of the horizontal shear stress on the bottom, τ_0 (force/area), to the mass density of water, ρ (mass/volume), i.e., $v_* = \sqrt{\tau_0/\rho}$, length/time

δ = height of a typical roughness element on the stream bottom, length

ν = kinematic viscosity of the stream water, area/time

A simple force balance on a short steady-state section of stream shows that τ_0 is (Yalin 1977):

$$\tau_0 = \gamma SR \tag{108}$$

where

γ = specific weight of the stream water, weight/volume

S = energy gradient of the stream, dimensionless

R = hydraulic radius, length

Recalling the Manning equation, it is clear that τ_0 is proportional to the square of the mean velocity, at least for steady flow. Since the derivation of Equation 108 assumes that the flow is unaccelerated, it does not strictly apply to the dynamic case at hand. However, Equation 108 does show that the shear velocity, v_*, depends on the mean velocity and should increase with it. Further analysis shows that v_* is correlated with both mean velocity and mean depth of flow, or hydraulic radius (Yalin 1977).

76. The net result of this analysis is that h_b is reduced if mean velocity increases or if hydraulic radius decreases. Thus, the flux density may be rewritten as:

$$\left\{\text{Flux density}\right\} = \frac{D_L \cdot g(U) \cdot C}{h(R)} \tag{109}$$

where

g(U) = monotonically increasing function of mean velocity

h(R) = monotonically increasing function of hydraulic radius

77. For use in a water quality model, flux density must be converted into a volumetric reaction rate. Clearly, the mass rate at which matter diffuses into the benthos is equal to its mass rate of disappearance from the water column. Therefore, for any fixed control volume, one can write:

$$\left\{\text{Flux density}\right\} \cdot \left\{\text{Bottom area}\right\} = \left\{\begin{matrix}\text{Reaction rate} \\ \text{per unit volume}\end{matrix}\right\} \cdot \left\{\text{Volume}\right\}$$

$$\left\{\begin{matrix}\text{Reaction rate} \\ \text{per unit volume}\end{matrix}\right\} = \frac{\text{Flux density}}{R}$$

$$\left\{\begin{matrix}\text{Reaction rate} \\ \text{per unit volume}\end{matrix}\right\} = -\frac{D_L \cdot g(U)}{R \cdot h(R)} \cdot C \tag{110}$$

The minus sign in the last equation indicates that the reaction is a sink. The fundamental definition of the hydraulic radius is the volume/area ratio.

78. Finally, it may be noted that both the steady-state mean velocity and hydraulic radius are simple power functions of the steady flow, Q (Leopold, Wolman, and Miller 1964). Therefore, the sink term can be written as a simple function of the discharge. One obvious but simple suggestion is:

$$\left\{\begin{matrix}\text{Rate of reaction} \\ \text{per unit volume}\end{matrix}\right\} = -aQ^b \cdot C \tag{111}$$

where

a,b = empirical coefficients

Q = stream flow, volume/time

aQ^b = first-order reaction rate coefficient, per time

The coefficients a and b must be regarded as site specific. Since b represents only the boundary layer thickness, it should be the same for all reactants. The coefficient a , however, includes the diffusivity of the reactant, so it may vary among reactants.

79. The removal of settleable particles is somewhat more complicated. The benthal boundary layer cannot be regarded as stable but must be thought of as being intermittently disrupted. Therefore, deposited particles are subjected to fluid drag forces. According to Yalin (1977), the mean velocity at which granular particles just begin to move is given by:

$$U_c = \sqrt{2.89g\left(\frac{\rho_s - \rho}{\rho}\right)d^{0.81}H^{0.19}} \tag{112}$$

where

g = gravitational acceleration, units

ρ_s = mass density of the particle, mass/volume

ρ = mass density of water, mass/volume

d = particle diameter, length

H = mean depth, length

At velocities less than that specified by Equation 112, granular particles remain at rest. Since in any river there is a distribution of particle sizes, any velocity must scour some particles, and high velocities must scour all

particles. Thus, as the discharge increases, the number of particles scoured must increase because U increases, and the net flux of settleable particles entering the benthos must decline. For the time being, subject to later revision, it is assumed that the net removal of settleable particles is describable by a law like Equation 111.

80. Finally there is the decay of soluble and nonsettleable material to be considered. It is assumed that this decay is substantially unaffected by turbulence, which is the classical Streeter-Phelps' (1925) assumption. Since decay by plankton is much slower than decay by benthos (presumably because benthos has much more biomass), in large rivers a lower bound must be set for the first-order decay rate given by aQ^b . This lower bound represents plankton decay, and it will be approximated by the bottle decay rates observed in the laboratory.

81. Some experimental support for this analysis is provided by Wright and McDonnell (1979), Garland (1978), and Kittrell and Furfari (1963). Wright and McDonnell's data for carbonaceous biochemical oxygen demand (CBOD) exertion are shown in Figure 6. In this case, there is a strong correlation between the CBOD decay rate, K_1 , and the stream discharge. The reported regression line is:

$$K_1 = 10.3\ Q^{-0.49} \tag{113}$$

where

K_1 = CBOD decay rate, per day, base e, 20° C

Q = stream flow, ft^3/sec

This equation was derived for stream depths between 0.9 and 32 ft* and stream flows from about 5 to 9,000 ft^3/sec. Equation 113 includes planktonic and benthal decay caused by both sedimentation and diffusion through the benthal boundary layer. It clearly supports the analysis just presented. The lower bound for K_1 may be taken from the data of Schroepfer, Robins, and Susag (1960), shown in Figure 7.

82. Ammonia is not settleable, but both planktonic decay and diffusion through the boundary layer into the benthos occur, so a formula for the

---

* A table of factors for converting non-SI units of measurement to SI (metric) units is presented on page 6.

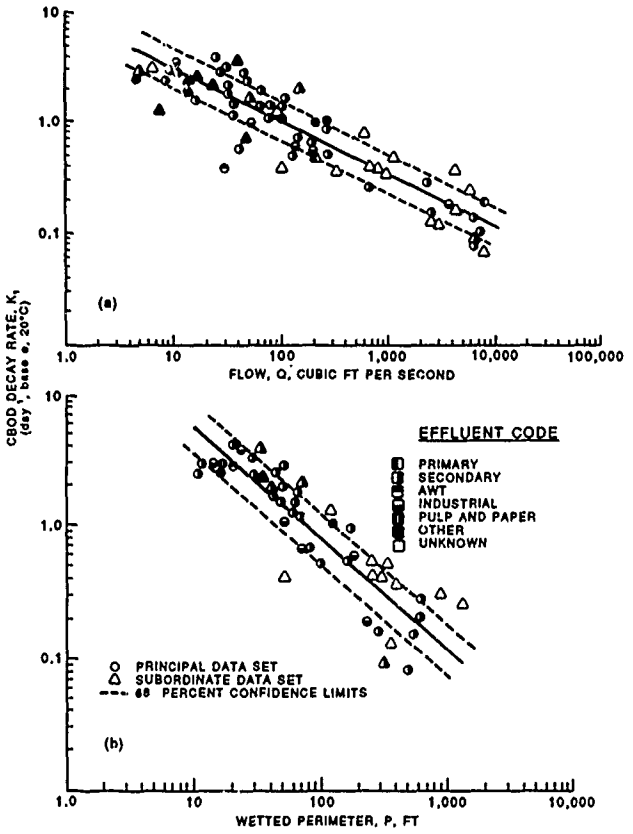


Figure 6. Variation of the stream K_1 with hydraulic properties (after Wright and McDonnell 1979) (Copyright 1979 American Society of Civil Engineers (ASCE), reproduced by permission of ASCE)

nitrogenous biochemical oxygen demand (NBOD) decay rate, K_n, like Equation 111 could be expected. The lower bound for K_n may be approximated by the bottle decay rates given in Table 1. These data clearly show the strong effect of water quality on K_n. Comparing Table 1 and Figure 7, it may be noted that bottle values for K_n are as much as 10 times the bottle values for K_1. Aside from this, it must be admitted that strong support for representing K_n by a formula like Equation 111 is lacking. Bansal's (1976) data for K_n are shown in Figure 8. These data are ambiguous. The regression line is:

$$K_n = 0.379 \cdot 10^{-3}\, g^{0.68}\, \nu^{-0.36}\, H^{0.04} \qquad (114)$$

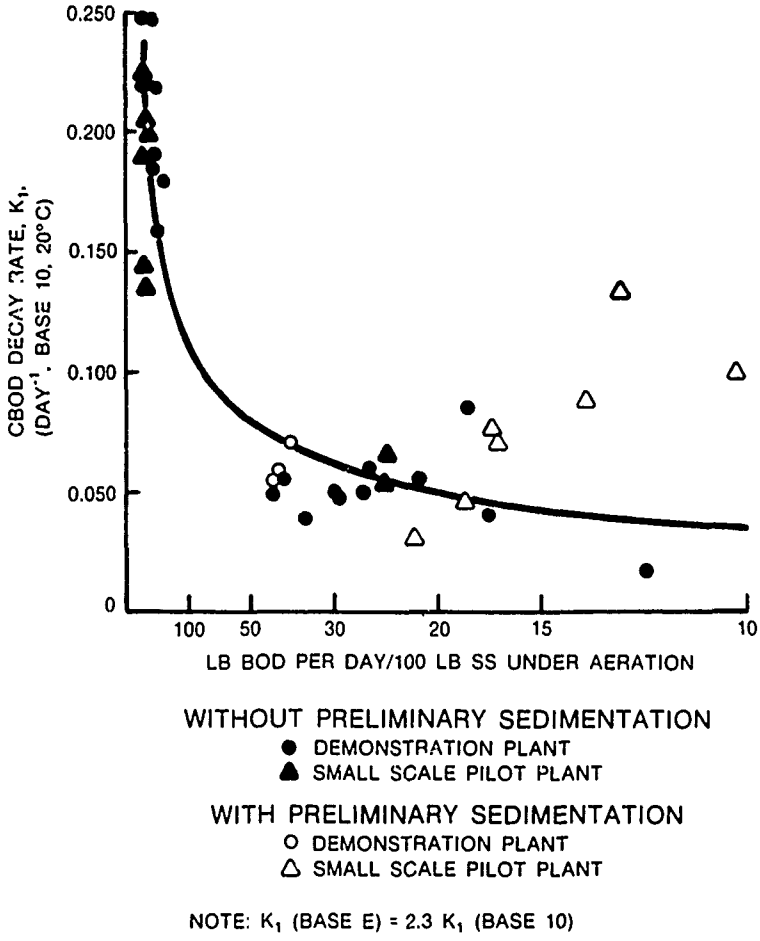


Figure 7. Variation of the bottle K_1 with the degree of biological wastewater treatment (after Schroepfer, Robins, and Susag 1960) (Copyright 1960 Water Pollution Control Federation (WPCF), reproduced by permission of WPCF)

where

K_n = NBOD decay rate, per day, base e, 20° C

g = acceleration because of gravity, ft/sec^2

ν = kinematic viscosity, ft^2/sec

H = stream depth, ft

Unfortunately, most of Bansal's data comes from moderately deep rivers, 5 to 10 ft, and his regression line yields a nearly constant K_n of about 0.25 per day (base e, 20° C) in all cases. His graph shows two sets of outliers, one from the Flint River, Michigan, and the other from the Big Blue River, Nebraska. These rivers are only 1 to 2 ft deep. The Flint River data comprise relatively high K_n values, about 2.0 per day, whereas the Big Blue

Table 1

First and Second Stage Bottle BOD Parameters for the Scioto River and the Woutherly Wastewater Treatment Plant, Columbus, OH*

Statistic**		Raw Municipal Sewage	Primary Effluent	Secondary Effluent	Scioto River
Mean values First stage Parameters	K_1	0.230	0.251	0.294	0.169
	L_0	293	254	104	21.7
	w	-0.77	-0.42	0.50	0.99
Coefficients of variation	K_1	0.528	0.616	0.790	1.116
	L_0	0.430	0.543	0.798	0.811
	w	2.7	3.9	4.2	1.9
Sample sizes		60	56	62	47
Mean values Second stage Parameters	K_n	0.226	0.402	0.400	0.599
	L_N	265	250	64.7	15.9
	w_N	12.1	11.1	9.61	8.68
Coefficients of variation	K_n	0.787	0.642	0.198	0.413
	L_N	0.312	0.461	0.399	0.215
	w_N	0.208	0.334	0.232	0.181
Sample sizes		5	7	4	8

* From Strand (1975).

** K_1, K_n = decay rate coefficient, per day, base e, 20° C.
L_0, L_N = CBOD, NBOD, mg O_2/l.
w, w_N = lag period, days.

River data comprise relatively low values, about 0.1 per day. The lack of correspondence between the nitrification rates in the Flint and Big Blue Rivers may be due to differences in bottom substrate, since streams with unstable bottoms cannot be expected to develop significant benthal populations. Alternatively, their water quality may differ greatly. As a result, Bansal's (1976) data show no effect of hydraulic conditions upon nitrification. Curtis, Durrant, and Harman (1975) and Garland (1978) have shown that in the River Trent system (UK) the numbers of planktonic nitrifiers are insufficient to account for more than an insignificant portion of nitrification observed and, consequently, nitrification in the Trent system must be a benthal process. The Trent and its tributaries are less than 1.9 m deep in

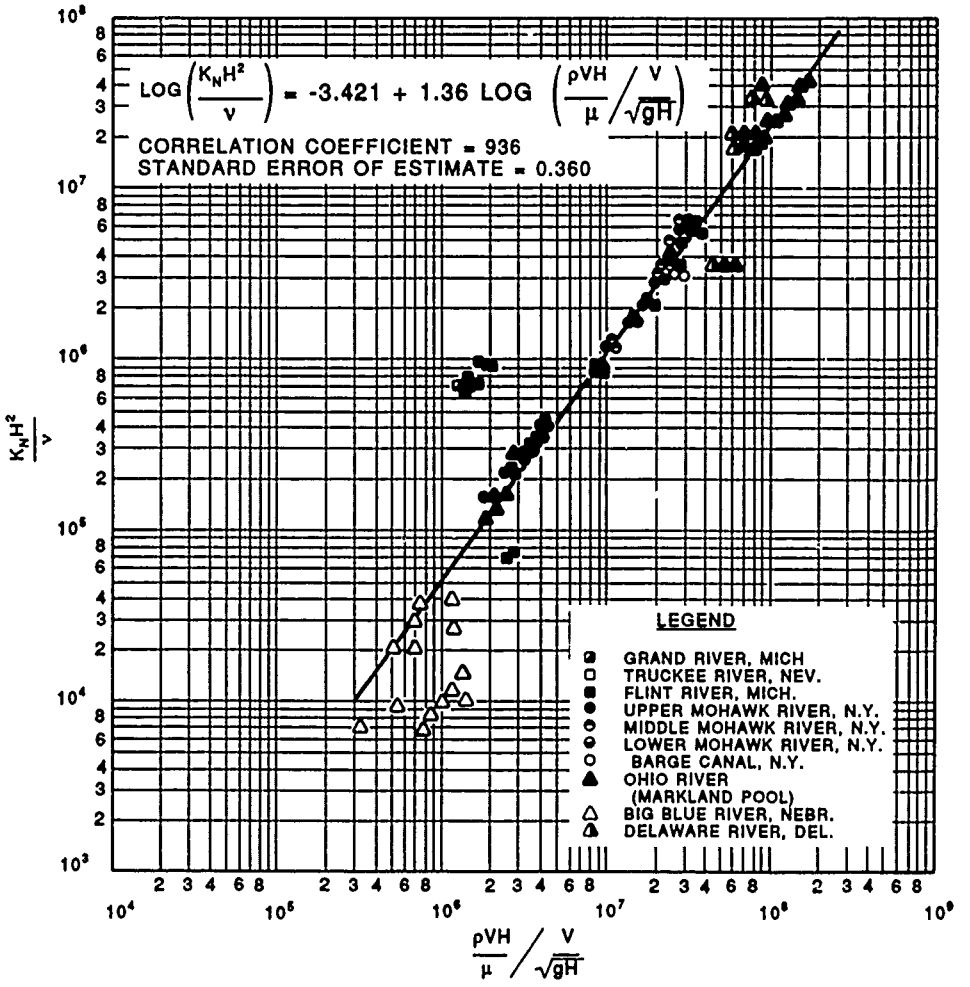


Figure 8. Variation of the river K_n with hydraulic properties (after Bansal 1976) (Copyright 1976 WPCF, reproduced by permission of WPCF)

nitrifying reaches, so benthal effects should be significant. Unfortunately, Garland (1978) also reported that the rate of nitrification was inversely related to the stream velocity:

$$K_n = 0.0833\ T + \frac{2.1234}{U} + 0.0646\ B_n$$

$$R^2 = 0.972 \qquad (115)$$

where

T = water temperature, °C

U = mean velocity, km/hr

B_n = concentration of Nitrosomonas, 1,000 cells/ml

Thus, the effects of turbulence upon nitrification remain uncertain. However, in view of the fact that nitrification occurs primarily in the benthos, it will be assumed that K_n can be represented as a function of the discharge like Equation 111.

83. Finally, Kittrell and Furfari (1963) have shown that coliform bacteria decay much faster in small than in large streams, and they attribute this to benthal predation upon the coliforms. Their conclusions are supported by a more complete study by Velz (1970), but the results, again, are not clear-cut. Velz's summary is reproduced in Table 2, and it will be noted that the Hudson River does not conform to Kittrell and Furfari's general rule. Nevertheless, since coliform decay appears to be a benthal process, the coliform decay rate will be assumed to vary with flow like K_1 in Equation 111.

Reaeration

84. The user has the option to use either Equation 111 or a constant for biodegradable decay rates. In either case, the decay rates are corrected for temperature.

85. Stream reaeration can be approached in a similar manner. Since planktonic decay and sedimentation are not directly involved, the analysis is concerned only with diffusion through the surface boundary layer. Churchill, Elmore, and Buckingham (1962) state that the relevant dimensional groups are:

$$\text{Stream Reynold's number} = \frac{\rho UH}{\mu} = N_R \tag{116}$$

$$\text{Molecular Schmidt number} = \frac{\mu}{\rho D_L} = N_{Scm} \tag{117}$$

$$\text{Turbulent Schmidt number} = \frac{\mu}{\rho E_v} = N_{Sct} \tag{118}$$

$$\text{Darcy-Weisbach friction factor} = f \tag{119}$$

Table 2

Coliform Death Rates K Observed in Rivers*

Chick's Law: $\text{Log}\ (B/B_o) = -Kt$

River	Reaction Rate K, day^{-1}, base 10		Authority for Survey Data**	Remarks
	Warm Weather	Cool Weather		
Ohio	0.50	0.45	Frost and Streeter (1924)	Generalized results of analysis of extensive data
Upper Illinois	0.90 0.67	0.32 0.29	Hoskins et al.	1-day decline 2-day decline
Scioto	0.96	0.46	Kehr et al.	
Hudson	0.80		Hall, Riddick, Phelps	Freshwater reach below Albany
Upper Miami	0.80		Velz, Gannon, Kinney	Mean through reach above Dayton
Tennessee	0.46		Kittrell	1- and 2-day declines, below Knoxville
Tennessee	0.60 0.57		Kittrell	1-day decline 2-day decline (below Knoxville)
Sacramento	0.77 0.65		Kittrell	1-day decline 2-day decline (below Sacramento)
Missouri		0.30 0.26	Kittrell	1-day decline 2-day decline (below Kansas City)

* Fron Velz (1970) (reproduced by permission of John Wiley & Sons, Inc., New York).

** Additional bibliographical information may be found in Velz (1970).

$$\text{Ratio of the Reynolds number to the Weber number} = \frac{\sigma}{\mu U} = \frac{N_R}{N_W} \tag{120}$$

$$\text{Sherwood number} = \frac{K_2 H}{U} = N_{sh} \tag{121}$$

where

ρ = water density

U = cross sectionally averaged stream velocity, length/time

H = stream hydraulic depth, length

μ = absolute viscosity, mass/length/time

D_L = molecular diffusivity of oxygen in water, length^2/time

E_v = vertical turbulent diffusivity, area/time

σ = surface tension, force/length

K_2 = reaeration rate, 1/time

Dimensional analysis indicates that the Sherwood number should be a function of the other groups. Churchill, Elmore, and Buckingham (1962) assumed this function could be written as a simple product:

$$K_2 = c \frac{U^a}{H^b} \left(N_R\right)^d \left(N_{Scm}\right)^e \left(N_{Sct}\right)^g \left(N_W\right)^i f^j \tag{122}$$

where a, b, c, d, e, g, i, j = empirical coefficients.

86. The dimensionless form of the Sherwood number was discarded in favor of the ratio $K_2 H^b / U^a$. Equation 122 and several modifications of it were fitted to an extensive set of field data by multiple regression. The rivers used were the Clinch, Holston, French Broad, Watauga, and Hiwassee, all of which are in the Tennessee Valley Authority (TVA) system. The mean depths ranged from 2.12 to 11.41 ft, and the mean velocities ranged from 1.85 to 4.65 ft/sec. The regression analysis indicated that f, N_R, N_{Scm}, N_{Sct}, and N_W had little influence on K_2 for these data, so that the preferred formula was:

$$K_2 = 11.61 \frac{U^{0.969}}{H^{1.673}} \; ; \; R^2 = 0.676 \tag{123}$$

where

K_2 = reaeration rate coefficient, per day, base e, 20° C

U = mean stream velocity, ft/sec

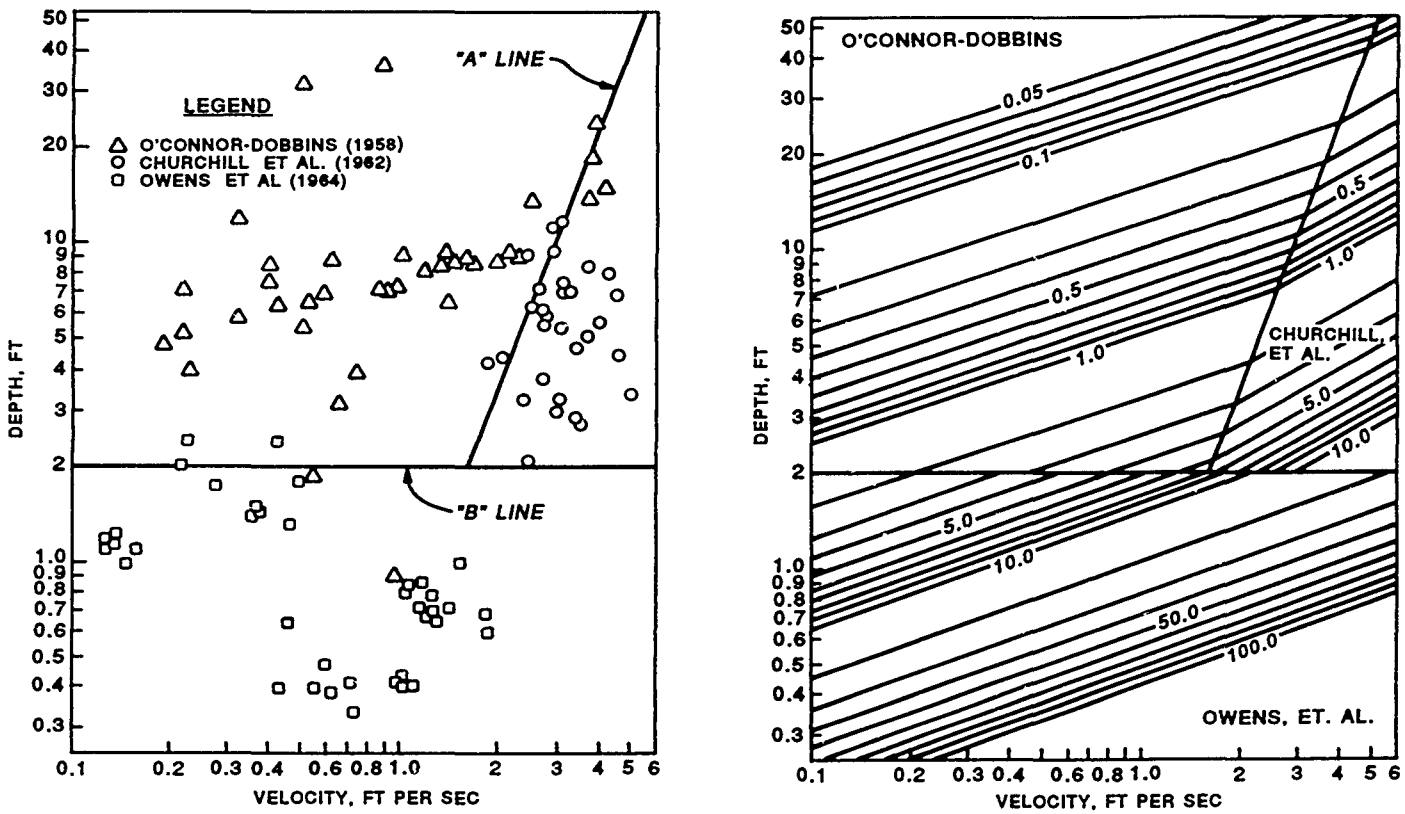
H = mean stream depth, ft

It may be noted that Equation 123 has the same form as the K_2 formula originally proposed by Streeter and Phelps (1925). The model was base e, so equations that compute K_2 in base 10 should be multiplied by 2.31 to obtain base e.

87. The only stream characteristics incorporated into Equation 122 are the mean velocity and depth; other variables are either water properties (ρ, μ, σ), solute properties (D_L), or derivatives of U and H (f, E_v). The water and solute properties function as mere temperature corrections. The one key variable neglected by Churchill, Elmore, and Buckingham (1962) is acceleration caused by gravity, g . If this variable is included, Equation 122 becomes a function of the Froude number ($N_{Fr} = U/\sqrt{gH}$), too. However, this does not change the fact that U and H are the only stream characteristics included.

88. It is clear that all K_2 theories must involve N_{Sh}, N_{Fr}, N_{Scm}, N_{Sct}, N_W and N_R and, consequently, that all theories must be reducible to a function of U and H . If the Streeter-Phelps formula for K_2 is adopted (e.g. Equation 123), then all local stream properties are incorporated in the exponents of U and H and the proportionality coefficient. These exponents and the coefficient thus become properties of individual stream reaches and must be expected to vary from one reach to the next. The very large number of K_2 formulas in use shows that this is indeed the case (Brown 1974; Wilson and Macleod 1974). Covar (1976) has recommended that each K_2 formula be used only in reaches that have the same velocity and depth as the reaches used to verify the formula. His recommendations for the use of three formulas are shown in Figure 9. Besides Equation 123, the formulas studied by Covar include the following from Owens, Edwards, and Gibbs (1964):

$$K_2 = 21.74 \frac{U^{0.67}}{H^{1.85}} \tag{124}$$



a. Data used in verification of K_2 formulas

b. Recommended K_2 values (per day, base e, 20° C)

Figure 9. Suggested guidelines for selection of K_2 formulas (after Covar 1976)

where

K_2 = reaeration rate coefficient, per day, base e, 20° C

U = mean velocity, ft/sec

H = mean depth, ft

and the following from O'Connor and Dobbins (1958):

$$K_2 = \frac{\sqrt{D_L U}}{H^{3/2}} \tag{125a}$$

or

$$\frac{K_2 H}{U} = \left(\frac{\rho D_L}{\mu}\right)^{1/2} \cdot \left(\frac{\mu}{\rho U H}\right)^{1/2} \tag{125b}$$

$$N_{Sh} = N_{Scm}^{-1/2} \cdot N_R^{-1/2} \tag{125c}$$

The O'Connor--Dobbins' equation reduces immediately to a version of Equation 122. Any consistent set of units is permissible in Equation 122. The calculated K_2 will be base e with the units chosen for time in U and D_L, and it will apply at the temperature chosen for D_L. The value of D_L is 0.0019 ft^2/day at 20.0° C. With this value for D_L and correcting for time units, Equation 125a becomes

$$K_2 = \frac{12.81\ U^{1/2}}{H^{3/2}} \tag{125d}$$

where K_2 is in units of per day, base e, at 20° C; U is in units of feet per second and H is in feet. The model applies stream reaeration equations like Equations 123, 124, and 125d in the form

$$K_2 = \frac{AG\ U^{E1}}{H^{E2}} \tag{126}$$

where the user inputs the values for E1 , E2 , and AG ; AG should have units of day^{-1} fps^{-E1} ft^{E2} for base e and 20° C. The model performs the conversion to SI units.

89. The statement that all reaeration rate formulas may be reduced to a form like Equation 122 includes two important examples that deserve separate mention. The first method (Velz 1939, 1947, 1970) is based on an analysis of the diffusion of oxygen into quiescent water (Black and Phelps 1911). Velz's method is graphical, because the Black-Phelps' solution expresses the rate of reaeration as a power series. However, Deininger (Velz 1970) has shown that a closed-form solution is possible, too. After some manipulation, the closed-form solution yields:

$$\text{Rate of absorption per unit volume} = 2\sqrt{\frac{D_L}{\pi H^2 t^*}}\,(C_s - C)$$

or, clearly:

$$K_2 = 2\sqrt{\frac{D_L}{\pi H^2 t^*}} \tag{127}$$

where

t^* = time between complete mixes

C_s = saturation concentration of oxygen, mass/length^3

C = concentration of oxygen, mass/length^3

The variable t^* is a function of U and H , although Velz assumes it is a function of H only. This is a reasonable assumption, since H is much more variable than U . It must be noted that Velz's empirical curves for t^* were derived using an incorrect value for D_L , and the use of his curves requires that his D_L value be used also.

90. The second important example is the Tsivoglou-Wallace (1972) equation:

$$K_2 = 0.054\,\frac{H_L}{\theta} \tag{128}$$

where

K_2 = reaeration rate coefficient, per day, base e, 20° C

H_L = total head loss along a reach, ft

θ = time of travel along the reach, days

The coefficient in Equation 128 has units of ft^{-1} and varies inversely with discharge. The value given is a typical average stream value (Tsivoglou and Wallace 1972). Churchill, Elmore, and Buckingham (1962) have shown that this is merely another version of Equation 122, as may be illustrated for wide shallow channels in the following example. From the Manning equation, one computes:

$$S = \frac{H_L}{\theta U} = \frac{n^2 U^2}{1.486^2 H^{4/3}} \tag{129}$$

where it has been assumed that H is nearly equal to R; S is the energy gradient; and the length of the reach is θU. Solving for H_L/θ and substituting into Equation 122 yields:

$$K_2 = \left(0.054 \frac{n^2}{1.486^2}\right) \frac{U^3}{H^{4/3}} \tag{130}$$

This is obviously another version of the Streeter-Phelps' formula.

91. Since all K_2 formulas are reducible to the form aU^b/H^c, they are also reducible to the form aQ^b, like K_1 and K_n. However, it is customary to leave the K_2 formulas in the first form.

92. The stream reaeration formulas discussed above account for gas transfer resulting from stream turbulence. Wind-generated turbulence at the air-water interface can also cause gas transfer to/from the atmosphere. To allow for this effect, a wind-driven reaeration formulation by O'Connor (1983) can be applied. The wind-driven gas transfer rate, R_k (L/T), is dependent upon wind speed, air temperature, and water temperature with parameters in the model set for small-scale water bodies. Dividing R_k by depth yields a reaeration coefficient like K_2 (1/T). Another popular wind-driven reaeration formulation is that of Banks (1975).

93. Surface reaeration can be augmented by structural reaeration as flow passes through control structures (dams). Structural reaeration is

computed with the energy dissipation model (EDM) (Wilhelms and Smith 1981), which states

$$D_f = D_i e^{-c\Delta h} \tag{131}$$

where

D_f = final DO deficit, after structural reaeration, mg/ℓ

D_i = initial DO deficit, before structural reaeration, mg/ℓ

c = escape coefficient, 1/ft

Δh = change in water surface elevation from upstream pool to tailwater, ft

The DO deficit, D, is expressed as

$$D = C_{sat} - C \tag{132}$$

where

C_{sat} = saturation concentration of DO, mg/ℓ

C = actual concentration of DO, mg/ℓ

94. Wilhelms and Smith (1981) recommended that for gated conduits the value of the escape coefficient at 20.0° C, c_{20}, should be set to 0.045 ft^{-1}. The idea of the EDM was borrowed from the Tsivoglou and Wallace (1972) reaeration model for streams. Comparing $c_{20} = 0.045\ ft^{-1}$ with the coefficient in Equation 128, it appears that structural and stream reaeration exhibit much similarity. There is a temperature correction of c (Equation 137).

95. Wilhelms and Smith (1981) suggest that flow, geometry, turbulence, and other factors can influence structural reaeration. These effects can be lumped into the escape coefficient. Therefore, c_{20} can vary among projects. Although the EDM was tested on gated conduits, it gives results similar to the Holler (1970) model for spillways. For hydropower structures, it would be reasonable to assume that little or no reaeration occurs as most of the energy is extracted for power generation. Caution should be exercised for structures that have a submerged jump in the tailwater, thus reducing reaeration.

Temperature

96. Streeter and Phelps (1925) suggested that the CBOD decay rates measured in bottles could be corrected for temperature by the following equation:

$$\frac{K_1(T)}{K_1(20)} = \theta_c^{T-20} \tag{133}$$

where

$K_1(T)$ = decay rate at temperature T

T = incubation temperature, °C

$K_1(20)$ = decay rate at 20° C

θ_c = an empirical constant

Equation 133 was shown by Theriault (1927) to be an approximation of the Arrhenius rate law. Consequently, θ_c should be a function of the reaction's activation energy, the gas constant, and the temperature. Fortunately, for small temperature ranges, θ_c is nearly constant. Streeter and Phelps (1925) reported that the average value of θ_c over the range 10° to 37.5° C was 1.047. Some of their data collected between 10° and 20° C yield a θ_c value of 1.0524, and other data collected between 24.5° and 37° C yield a value of 1.01 and 1.135 for 4° to 20° C. Theriault reported θ_c values of 1.053 and 1.049 for the temperature ranges 9° to 20° C and 20° to 30° C, respectively, and concluded that the mean value reported by Streeter and Phelps was adequate for practical purposes: i.e., 1.047.

97. Equation 133 can be extended to cover nitrification as well:

$$\frac{K_n(T)}{K_n(20)} = \theta_N^{T-20} \tag{134}$$

Parker et al. (1975) and Zison et al. (1978) have published reviews indicating that the bottle values of θ_N range from about 1.055 to about 1.12 for nitrification. The mean of the reported values is about 1.09, which seems as good an estimate as any.

98. Finally, K_2 must also be corrected for temperature, unless the O'Connor-Dobbins' or Black-Phelps' formulas are used; in these cases, the temperature correction is incorporated in D_L. The customary correction formula for K_2 is:

$$\frac{K_2(T)}{K_2(20)} = \theta_o^{T-20} \tag{135}$$

The usual value chosen for θ_o is 1.024 (Elmore and West 1961). The diffusivities of oxygen in water at 10°, 20°, and 30° C are 1.38×10^{-5}, 2.037×10^{-5}, and 2.85×10^{-5} cm^2/sec, respectively (Camp and Meserve 1974). Substitution of these values into the O'Connor-Dobbins' formula yields an effective θ_o of 1.020 between 10° and 20° C and 1.017 between 20° and 30° C. Velz (1970) recommends that the diffusivity be computed as:

$$D_L = (2.04 \times 10^{-5}) \cdot 1.04^{(T-20)} \tag{136}$$

where D_L is the diffusion coefficient of oxygen in water in square centimetres per second. The Velz correction yields a value of 1.04 for θ_o.

99. Reaeration accomplished by hydraulic structures is also affected by temperature. Wilhelms and Smith (1981) suggest that the escape coefficient, c, be corrected for temperature by the following equation:

$$c(T) = c_{20}\ 1.02^{(T-20)} \tag{137}$$

PART VI: SCHEMATIC MODEL OF RIVER WATER QUALITY

100. In the following paragraphs, the mathematical formulations of the sources and sinks and reaction kinetics for various water quality components are presented. The most significant of these are temperature and DO. However, other variables are also included because of their effect on DO. These variables are CBOD and organic and ammonia-nitrogen (nitrification). The growth and decay of algae and macrophytes are also modeled to complete the DO balance, but algae and macrophytes are not state variables. Nitrite plus nitrate-nitrogen is included to complete the nitrogen cycle. Phosphate-phosphorus is modeled as a nutrient, and its concentration is a good indicator of stream pollution. Coliform bacteria are also included as a modeled variable because of their importance as an indicator of stream pollution. The coliform bacteria variable can be used to model another nonconservative variable or a conservative variable (by setting the die-off rate to zero). Dissolved iron and manganese are included because of their impact in streams below dams with anoxic hypolimnetic releases. Thus, 10 modeled variables are included; the ones to be modeled are selected by the user. Each variable must be transported by Equation 76. Figure 10 shows schematically how the variables interact.

Temperature

101. It is necessary to account for the effects of temperature changes along the channel. Either cultural inputs such as power plant effluents or natural processes are responsible for such potential gradients. In addition to its intrinsic value, temperature information is required for specification of corrections to rate coefficients. Temperature may be either read in from existing data, thereby precluding model predictability, or it may be calculated/predicted from the solution of an appropriate thermal energy transport equation.

102. The transport equation (Equation 76) developed in Part IV, is used to solve for each water quality variable, α ; thus for temperature, $\alpha \equiv T$. The specification of the source/sink term and the decay term of Equation 7 must account for all the mechanisms other than advection, diffusion,

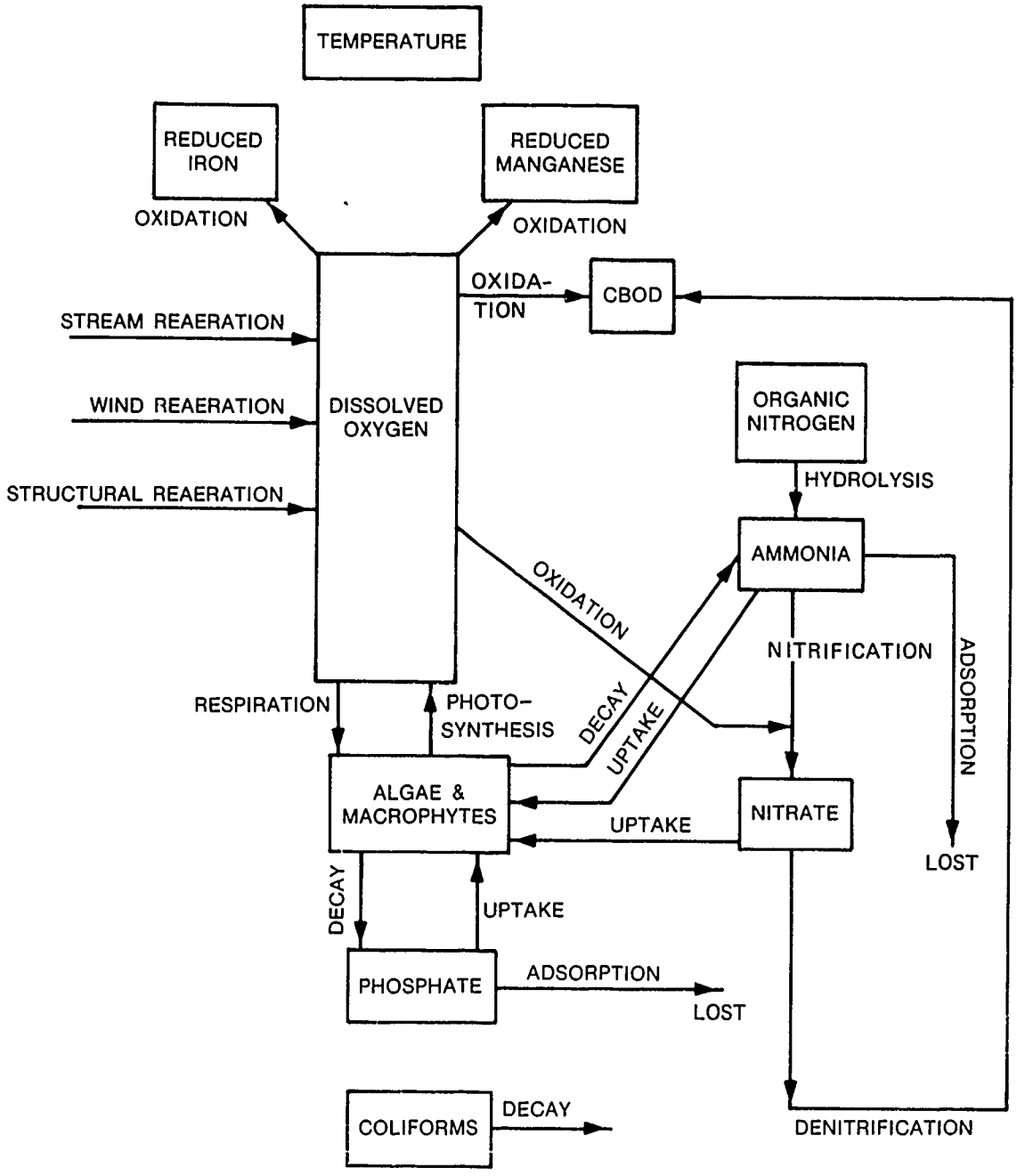


Figure 10. CE-QUAL-RIV1 water quality compartmental diagram

and lateral inflows, which cause net transfers of energy to the control volume.

103. In general, the external sources and sinks (excluding lateral inflows) for heat are described by

$$H_N = H_S + H_L - H_E - H_B \pm H_C \tag{138}$$

where

H_N = net heat transfer, $\frac{\text{heat energy}}{\text{surface area} \cdot \text{time}}$

H_S = net short-wave radiation

H_L = net long-wave radiation

H_E = heat loss because of evaporation

H_B = heat loss because of back radiation of the water

H_C = heat transferred by conduction at the water surface and the bottom

104. There are two options for computing H_N. In the first option, each term in Equation 138 is computed (i.e., subroutine HEATFLUX) and added to form H_N, which is converted to a rate of temperature change by

$$\dot{\Delta T} = \frac{H_N}{\rho C_p H} \cdot \text{conv} \tag{139}$$

where $\dot{\Delta T}$ = rate of temperature change, degrees/time

ρ = specific mass of water, $\frac{\text{mass}}{\text{vol}}$

C_p = specific heat of water, $\frac{\text{heat energy}}{\text{mass} \cdot \text{degree}}$

H = hydraulic depth, $\frac{A}{B}$, length

conv = conversion factor from English to metric units (HEATFLUX performs computations in English units)

The variable $\dot{\Delta T}$ is used for the SINKS term in Equation 76, and the K_S term is zero. The algorithms in subroutine HEATFLUX are based on those used in QUAL-II (Roesner, Giguere, and Evenson 1981) to compute the terms in Equation 138. The computations within HEATFLUX depend on: water temperature, T_S; time of year and day, site latitude, longitude, and elevation; and local meteorological data. The meteorological data, obtained from the National Oceanic and Atmospheric Administration (NOAA), consist of dry and wet bulb (or

dew point) temperatures, wind speed, cloud cover, and barometric pressure. The previously computed value (previous time-step) is used for T_S . Generally, meteorological data are collected and reported in 1- or 3-hr intervals. This method is referred to as a direct energy balance because determination of the full terms on the right side of Equation 138 is made within the simulation in contrast to the alternative heat exchange method.

105. The alternative method uses the well-known equilibrium temperature approach, as developed by Edinger, Duttweiler, and Geyer (1968) and Edinger, Brady, and Geyer (1974), to account for the effects of surface heating and radiation. The equilibrium temperature approach is based on the concept that heat exchange does not occur when the water temperature, T_S , equals the equilibrium temperature, T_E ; thus, $H_N = 0$. With $T_S = T_E$ and $H_N = 0$, the terms on the right side of Equation 138 can be expressed in terms of T_E , and T_E can be solved iteratively. Net heat transfer, H_N , is computed from

$$H_N = K_E (T_E - T_S) \tag{140}$$

where K_E is the heat exchange coefficient (heat energy/surface area/time/degree). The variable K_E can also be solved through manipulation of Equations 138 and 140. With T_E and K_E known a priori and using T_S from the previous time-step, H_N (and $\dot{\Delta}T$) can be determined. Using a computer program with the same meteorological data used for the direct energy balance, T_E and K_E are computed independent of the simulation. This program can be obtained from WES or the Hydrologic Engineering Center (HEC). A description of the T_E and K_S computation is given in Edinger, Brady, and Geyer (1974).

106. If the equilibrium temperature approach is used, the model uses a single, constant value for T_E and K_E . Thus, temperature is modeled with a constant meteorological forcing. The computations of T_E and K_E in the WES and HEC program are based on daily average values for meteorological data; thus daily average values of T_E and K_E are produced. Daily average values for T_E and K_E are adequate for reservoir thermal simulations but may not be adequate for riverine applications where diel fluctuations are important. In these cases, the direct energy balance method should be used. The T_E and K_E program can be modified to calculate T_E and K_E values that are commensurate with the meteorological data update interval (i.e., 1- or 3-hr intervals).

107. By inspection of Equation 140, the K_S coefficient of Equation 76 is K_E (KTS in the code) and the SINKS term is $K_E T_E$ (KTS*TEQ in code). Thus, KTS is converted to units of per time (per day) by dividing by (ρcpH) with the proper conversion units.

108. Either heat exchange method can be augmented with bottom heat transfer by adding the term K_B (TSINK - T_S) where TSINK is the bottom/ground temperature in degrees and K_B is the bottom heat transfer coefficient in units of per day. In this case, K_B and TSINK must be specified, the K_S term of Equation 76 is incremented by K_B (KTB in the code), and the SINKS term of Equation 76 is incremented by K_B*TSINK .

109. Future developments for this model should include incorporating mechanisms to allow for shading as this effect can be significant for small streams. Additionally, consideration should be given to allow the solar radiation that passes through the water column (for clear, shallow streams) to be absorbed by the bottom, thus providing bottom warming later. At present, all solar radiation is absorbed by the water column.

Carbonaceous Biochemical Oxygen Demand

110. The CBOD represents (as oxygen equivalents) the amount of biodegradable organic matter present, not counting organic nitrogen. The model calculates on the basis of the ultimate first-stage biochemical oxygen demand (BOD) (i.e., CBOD), and input to the model must also be CBOD's. If BOD_5 data are available, these must be converted to CBOD by the following formula:

$$CBOD = \frac{BOD_5}{1 - \exp(-5K)} \tag{141}$$

where

CBOD = ultimate first-stage (carbonaceous) BOD, g/m^3

BOD_5 = 5-day (carbonaceous) BOD, g/m^3

K = bottle BOD decay rate, day^{-i}

The BOD_5 should be determined using nitrification-inhibited samples to avoid double counting of the NBOD. If bottle K values are not available for the wastewater at hand, they may be approximated using Figure 7.

111. The CBOD is oxidized by the heterotrophic flora of the benthos using either oxygen or nitrate as the terminal electron acceptor. The latter process is called denitrification. In this model, it is assumed that oxygen is the terminal electron acceptor at high DO's and nitrate at low DO's. Thus, the basic reaction for CBOD is (in FORTRAN for convenience):

$$\left\{\begin{array}{c}\text{Rate of CBOD} \\ \text{oxidation}\end{array}\right\} = -(K1 + KDN)*CBODNS \qquad (142)$$

where

K1 = rate coefficient for aerobic oxidation of CBOD, day^{-1}

KDN = rate coefficient for nitrate reduction and anaerobic CBOD oxidation, day^{-1}

CBODNS = concentration of nonsettleable CBOD, g O_2/m^3

In the present model, only the nonsettleable CBOD is considered explicitly. The removal and oxidation of settleable CBOD and the formation and scour of sludge deposits could be considered in future modifications. However, the previous discussion on the effects of turbulence suggests that CBODNS may be regarded as the total CBOD as long as localized sludge deposits do not form. It should be realized that K_S is K1 + KDN , SINKS is zero, and α is CBODNS in Equation 76 by inspection of Equation 142.

112. The rate coefficients K1 and KDN are presumed to be functions of the degree of turbulence, the ambient water temperature, the local DO, and the local nitrate concentration. The optional turbulence dependence is taken from Wright and McDonnell (1979), with the additional assumption that all rate coefficients depend on the same power of the flow. The temperature correction is taken from Streeter and Phelps (1925), as is customary although arguable. The coefficient K1 is assumed to increase with DO according to a simple Monod function (cf. Hoover and Porges 1952), and the coefficient KDN is assumed to decrease according to an analogous formula. The coefficient KDN is also assumed to deperd on nitrate in the same way that K1 depends on oxygen.

$$K1 = (AK1*Q**BK)*(1.047**(TEMP-20.))*(DO/(DO+KOCB1)) \quad (143)$$

$$KDN = (ADN*Q**BK)*(1.047**(TEMP-20.))* (KOCBDN/(DO + KOCBDN))*(NO3N/(NO3N + KNCBDN)) \quad (144)$$

where

AK1, ADN = empirical coefficients reflecting the diffusivity and reactivity of the CBOD ($day^{-1} \cdot (m^3/day)^{-BK}$). If the turbulence option is not used, then the units of reactivity are day^{-1}.

Q = stream flow, m^3/day

BK = empirical exponent reflecting the thickness of the benthal boundary layer, dimensionless

TEMP = T_S = the ambient stream temperature, °C

DO = local stream oxygen concentration, g O_2/m^3

KOCB1 = Monod half-velocity constant for oxygen-limited aerobic systems, g O_2/m^3

KOCBDN = denitrification inhibition half-velocity constant g O_2/m^3

NO3N = local nitrate-nitrogen concentration, g N/m^3

KNCBDN = Monod half-velocity constant for nitrate-limited denitrification, g N/m^3

If the turbulence option is not used, then the term Q**BK is set to 1.0. No provision for fermentation has been provided, so in the absence of oxygen and nitrate, there is no CBOD removal.

Forms of Nitrogen

113. The forms of nitrogen (N) recognized by the model are organic nitrogen, ammonia, and nitrate. Nitrite is not considered because the overall rate of nitrification is ammonia limited (Parker et al. 1975), and stream surveys do not discover significant increases in nitrite in nitrifying reaches (Garland 1978, Miller and Jennings 1979).

Organic nitrogen

114. Organic nitrogen is a constituent of the organic matter that gives rise to the CBOD, although it is not part of the CBOD itself. For pure compounds, it is well known that their constituent nitrogen is released as

ammonia during the exertion of the CBOD. This release suggests that the rate coefficient for the conversion of organic nitrogen to ammonia is simply K1 or KDN :

$$\left\{\begin{array}{l}\text{Rate of hydrolysis} \\ \text{of organic nitrogen}\end{array}\right\} = -(K1+KDN)*ORGAN \qquad (145)$$

where, ORGAN is the concentration of organic nitrogen, grams of nitrogen per cubic metre (g N/m^3).

115. Although Equation 145 is conceptually appealing and is actually used in the model, it may be questioned whether the measured organic nitrogen in streams undergoes any net removal. For example, Ruane and Krenkel (1977) have published data for the Holston River that exhibit little change in organic nitrogen even in rapidly nitrifying reaches. Therefore, in some field applications organic nitrogen may be regarded as inert and deleted from the model. If organic nitrogen hydrolysis does occur, the model computes the hydrolysis rate coefficients from Equations 143 and 144.

Ammonia

116. Ammonia increases because of (a) the hydrolysis of organic nitrogen (if it occurs) and (b) algal and macrophyte decay. It decreases because of (a) nitrification, (b) algal and macrophyte uptake, and (c) ion exchange.

117. The increase in ammonia because of organic nitrogen hydrolysis is represented by Equation 145, with the minus sign changed to a plus because the process is a source of ammonia.

118. The rate of increase resulting from algal and macrophyte decay is taken to be proportional to the rate of decay of the plants. The nitrogen content of algae is variously reported as 7 to 10 percent by weight (Foree and McCarty 1968, MacKenthum and Ingram 1967), whereas the nitrogen content of macrophytes is only 2 to 4 percent by weight (MacKenthum and Ingram 1967, Gerloff 1969). Assuming algae are dominant and a nitrogen content of 7.5 percent (Foree and McCarty 1968):

$$\left\{\begin{array}{l}\text{Rate of increase of ammonia-N} \\ \text{caused by algal decay}\end{array}\right\} = +0.075*ALGADK \qquad (146)$$

where ALGADK is the rate of algal decay, grams of algae per day per cubic metre (g algae/day/m^3). It may be noted that all the algal N is assumed to be released upon decay of the cell (cf. Foree and McCarty 1968, Jewell and McCarty 1968). The algal decay rate itself is considered in greater detail in the following paragraphs.

119. The rate of ammonia decrease because of nitrification is assumed to be first order:

$$\left\{\begin{array}{l}\text{Rate of nitrification}\\ \text{of ammonia-N}\end{array}\right\} = -KN*NH3N \qquad (147)$$

where

KN = nitrification rate coefficient, day^{-1}

NH3N = ammonia-N concentration, g N/m^3

The rate coefficient KN is assumed to depend on turbulence (optional), oxygen, and temperature in a way similar to K1 :

$$KN = (AKN*Q**BK)*(1.1**(TEMP-20))*(DO/(DO+KON)) \qquad (148)$$

where

AKN = empirical coefficient reflecting the diffusivity and reactivity of ammonia, $day^{-1} \cdot (m^3/day)^{-BK}$; if the turbulence option is not used, then the units of reactivity are day^{-1} and Q**BK equals 1.0

KON = Monod half-velocity constant for oxygen limitation of nitrification, g O_2/m^3

The form of the oxygen limitation factor is suggested by the data summarized by Parker et al. (1975). Considering the reported variance in the temperature correction factor, θ_N , it has been rounded to two significant figures. A correction for pH has not been incorporated because of the width of the optimum range (about pH 7 to 9 according to Parker et al. (1975)). This feature might have to be changed if the model is to be applied to soft water or acid streams.

120. The rate of uptake of ammonia nitrogen caused by algal and macrophyte growth is entirely analogous to Equation 146. However, in this case it is assumed that the plants can use both nitrate and ammonia. The total

nitrogen consumption rate is partitioned between these two forms in proportion to their relative concentrations:

$$\left\{\begin{array}{l}\text{Rate of ammonia-N decrease} \\ \text{due to algal uptake}\end{array}\right\} = -0.075*(NH3N/(NH3N+NO3N))*ALGRO \quad (149)$$

where

NO3N = concentration of nitrate nitrogen, g N/m^3

ALGRO = reproduction rate of algae, g algae/day/m^3

The partitioning function was introduced primarily to avoid double counting of nitrogen uptake by plants. More sophisticated selectivity factors (e.g., O'Connor, Thomann, and DiToro 1973; Baca and Arnett 1976; DiToro et al. 1977) were not used, because there does not appear to be any empirical data or theory that warrants the use of such factors in models that lump all the photosynthetic flora together. This is not meant to deny that nutrient selection occurs in some pure cultures.

121. Finally, a term for physicochemical sorption of ammonia by the sediments is included. The need for such a process is shown by the data of Ruane and Krenkel (1977), Donigan and Crawford (1979), and Miller and Jennings (1979). The first two reports recorded losses in total inorganic nitrogen along well-aerated nitrifying reaches of about one-third to two-thirds of the total input. Ruane and Krenkel (1977) also produced plant biomass data that suggested nitrogen uptake by plants was relatively small and inadequate to account for the observed losses. The diel-average oxygen levels in all three studies ranged from 2 to 8 mg/ℓ in the reaches studied, so denitrification may not have been substantial either. Moreover, sediment analyses reported by Donigan and Crawford (1979) show that in at least two small streams most of the ammonia in the water-sediment system is in the sediments.

122. It may be noted that most nitrification studies report only ammonia losses, and the computed KN in these studies is derived only from the ammonia data. It is clear that these KN's include all effects of nitrification, algal uptake, and ion exchange and that the amount of nitrification occurring is generally overstated.

123. The representation adopted for ammonia sorption by sediment is a simple first-order decay, like that for CBOD:

$$\left\{\begin{array}{l}\text{Rate of sorption of} \\ \text{ammonia-N by sediments}\end{array}\right\} = -KNX*NH3N \qquad (150)$$

where KNX is the sorption rate coefficient, per day. The rate coefficient KNX is presumed to be a function of both turbulence (optional) and temperature.

$$KNX = (AKNX*Q**BK)*(1.025**(TEMP-20.)) \qquad (151)$$

where

AKNX = empirical coefficient reflecting the diffusivity and reactivity of ammonia, $day^{-1} \cdot (m^3/day)^{-BK}$; if the turbulence option is not used, then the units of reactivity are day^{-1} and Q**BK equals 1.0.

124. The combined ammonia-N reactions may be written as

$$\left\{\begin{array}{l}\text{Rate of accumulation of} \\ \text{ammonia-N due to reaction } (gN/m^3/day)\end{array}\right\} = (K1+KDN)*ORGAN+0.075*ALGADK$$

$$-0.075*(NH3N/(NH3N+NO3N))*ALGRO-(KN+KNX)*NH3N \qquad (152)$$

In Equation 152, the coefficient of the last term is K_S and the remaining terms make up SINKS of Equation 76.

Nitrate

125. Nitrate is formed by nitrification and removed by denitrification and plant uptake. Thus, the relevant formulas are Equations 142 (less K1), and Equation 147 (with the sign reversed), plus a term incorporating ALGRO :

$$\left\{\begin{array}{l}\text{Rate of accumulation} \\ \text{on nitrate-N } (gN/m^3/day)\end{array}\right\} = -0.35*KDN*CBODNS+KN*NH3N$$

$$-0.075*ALGRO*(NO3N/(NH3N+NO3N)) \qquad (153)$$

All terms in Equation 153 are in the SINKS term of Equation 76 and K_S is zero.

126. The coefficient 0.35 is a conversion factor for oxygen to nitrogen equivalents and is evaluated as follows. The half cell reaction for nitrate reduction is

$$6H^+ + NO_3^- + 5e^- = \frac{1}{2}N_2 + 3H_2O \quad (154)$$

Therefore, the equivalent weight of nitrate nitrogen is 2.8 g. The equivalent weight of oxygen is 8.0 g, and the mass of nitrogen equivalent to 1 g of oxygen is 0.35 g.

127. Wheatland, Barrett, and Bruce (1959) presented some interesting data regarding nitrification and denitrification in the Thames Estuary. First, denitrification occurred only in reaches with a DO below about 1 mg/ℓ, a finding they supported with laboratory data. More importantly, however, they showed that in the Thames the total inorganic nitrogen concentration declined from about 9 to about 3 mg/ℓ in a 15-mile reach nearly devoid of both oxygen and nitrate. While this decline might be interpreted as algal uptake, the depth and turbidity of the Thames argue against it, and the observed nitrogen losses might be due to simultaneous nitrification and denitrification at very low DO.

128. More recently, the studies of Williams and Lewis (1986) support the idea that nitrification and denitrification occur simultaneously because of processes within the sediments. Additionally, Hill and Sanmugadas (1985) showed that denitrification rates were significantly correlated to stream-sediment characteristics; this correlation helps to explain why recent studies have revealed considerable nitrate loss even during transport in well-oxygenated streams. Thus, denitrification via the water-sediment interface should be considered as a future development for this model.

Algae and Macrophytes

129. For convenience, all algae and macrophyte processes are lumped together. Their effects on state variables are included, but they are not modeled as state variables. Furthermore, the plants are assumed to be benthal and light-limited. This last assumption is defensible for small streams overhung by forest canopy and for turbid streams, both of which are common. However, the user can optionally elect to prevent algal/macrophyte growth if phosphorus or nitrogen are absent from the water. If the user wishes to use this simplistic nutrient limitation, then statements identified in the code must be commented/uncommented.

130. The plant growth rate, ALGRO (g algae/m^3/day), is taken to be the product of the benthal algal/macrophyte density, the local light intensity (corrected for attenuation en route), and a reaction rate coefficient. To further simplify the analysis, the stream cross section is approximated as a rectangle, as shown in Figure 11. Thus, the plant growth is distributed along the channel bottom and sides.

$$\left\{\begin{array}{l}\text{Plant growth per} \\ \text{unit length of channel}\end{array}\right\} = \left\{\begin{array}{l}\text{Bottom} \\ \text{growth}\end{array}\right\} + \left\{\begin{array}{l}\text{Side} \\ \text{growth}\end{array}\right\} \quad (155)$$

$$\begin{aligned}\text{ALGRO} = &\ (\text{B*KALGRO*SWALG*EXP(-KEXT*H)*ALGAEB} \\ &+ (2./\text{KEXT})*\text{KALGRO*SWALG*(1.-EXP(-KEXT*H))ALGAEB})/\text{A}\end{aligned} \quad (156)$$

where

B = stream top width, m

KALGRO = plant growth rate coefficient, m^2/watt/day

SWALG = light intensity (net short-wave radiation) at the water surface, watt/m^2

KEXT = light extinction coefficient for the particular reach, m^{-1}

H = hydraulic depth, m

ALGAEB = plant density on the bottom, g/m^2

A = channel cross-sectional area, m^2

The first term on the right-hand side of Equation 156 includes the light intensity on the bottom (SWALG*EXP(-KEXT*H)). The second term is the summation of the plant growth occurring at all depths on both (vertical) sides:

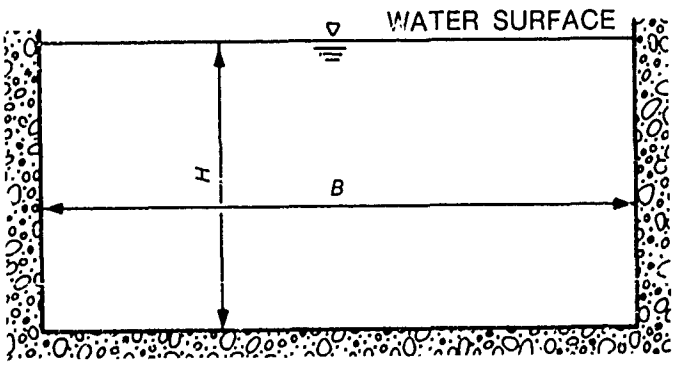


Figure 11. Channel geometry assumed for calculation of algal/macrophyte growth/decay

$$ALGRO_{sides} = 2.*KALGRO*ALGAEB*SWALG*\int_0^H EXP(-KEXT(H-Z))*dZ \qquad (157)$$

To account for diel effects, the surface light intensity (net short wave radiation) is computed as a function of the time of day. This is accomplished in subroutine HEATFLUX if the direct heat balance option is used; otherwise, the following relation is used:

$$SWALG = HNEFSW*SIN(PI*(CLOCK-DAWN)/LAMBDA) \qquad (158)$$

where

HNEFSW = maximum surface light intensity at local noon, watts/m^2

PI = 3.14159...

CLOCK = actual time of day, 24-hr clock

DAWN = time of local dawn, 24-hr clock

LAMBDA = elapsed time between local dawn and local sunset, hr

131. The elimination of nutrient effects on ALGRO is supported by the studies of Beck and Young (1975) on the River Cam in England. By introducing a term into the oxygen balance equation proportional to the total hours of sunlight, they were able to closely simulate an 80-day time series of oxygen data. They did not consider short-term variations of less than 24 hr. By comparison, a traditional Streeter-Phelps model without an "algal" term did not simulate the same field data as well. Moreover, Beck and Young (1975) also found it necessary to introduce a similar sunlight term into the BOD_5 balance equation, which suggests that "algae" manufacture BOD as well as DO. The present model, however, does not include an algal contribution to the stream's CBOD.

132. As Beck and Young (1975) noted, streams subjected to wastewater discharges are likely to have high nutrient concentrations and are also likely to be turbid. Both these conditions are reversed in lakes, so that the lake plankton are often, if not usually, nutrient- rather than light-limited. Thus, lake models almost always include nutrients in the formula for the algal growth rate. Of course, macrophytes obtain their nutrients from the sediments (Bole and Allan 1978, Carignan and Kalff 1980), so stream nutrient levels are irrelevant to macrophyte growth.

133. A rather simplistic nutrient limitation procedure is included in the model code, if the user elects to use it. This procedure sets ALGRO = 0.0 for segments void of phosphorus or nitrogen in the water. Future code improvements may allow implementation of a more rigorous nutrient limitation procedure.

134. Algae and macrophytes are presumed to decay continuously. The decay is supposed to cease at low DO levels. Because of the lack of reasonably precise data, no temperature correction is attempted. The decay rate is represented as:

$$\text{ALGADK} = \text{KALGDK}*\text{ALGAEB}*(\text{B}+2.*\text{H})/(1.+\text{KOALDK}/\text{DO})/\text{A} \qquad (159)$$

where

KALGDK = plant decay rate, day^{-1}

KOALDK = Monod half-velocity constant for oxygen limitation of plant decay, g O_2/m^3

Oxygen

135. The sources of oxygen (O) are reaeration and photosynthesis; the sinks are the exertion of CBOD, nitrification, plant respiration, and the oxidation of reduced iron and manganese. The present model represents plant photosynthesis and decay as algal photosynthesis and decay. The stoichiometry of these processes is based on the data of Foree and McCarty (1968) and depends on whether ammonia or nitrate is involved.

$$132\ CO_2 + 16\ NH_3 + H_3PO_4 + \frac{177}{2}\ H_2O = C_{132}H_{228}O_{58}N_{16}P + \frac{597}{4}\ O_2 \qquad (160)$$

$$132\ CO_2 + 16\ HNO_3 + H_3PO_4 + \frac{209}{2}\ H_2O = C_{132}H_{228}O_{58}N_{16}P + \frac{725}{4}\ O_2 \qquad (161)$$

The oxygen production indicated by Equation 160 is 1.59 g O_2/g algae, and that indicated by Equation 161 is 1.94 g O_2/g algae. Therefore, the rate of oxygen production by photosynthesis depends upon the relative proportions of ammonia and nitrate consumed:

$$\left\{\begin{array}{l}\text{Rate of oxygen production}\\ \text{by photosynthesis per unit}\\ \text{volume}\end{array}\right\} = +(1.59+0.35*(NO3N/(NO3N+NH3N)))*ALGRO \qquad (162)$$

136. The rate of oxygen consumption resulting from algal decay does not involve nitrification, so it is the reverse of Fquation 160:

$$\left\{\begin{array}{l}\text{Rate of oxygen consumption by}\\ \text{plant decay per unit volume}\end{array}\right\} = -1.59*ALGADK \qquad (163)$$

137. The rate of oxygen consumption resulting from CBOD decay is equal to the rate of CBOD decay, because of the definition of CBOD. This rate is given by Equation 142 (without KDN), and K1 is computed from Equation 143.

138. The rate of oxygen consumption resulting from ammonia oxidation is based on Equations 147 and 148. Since the units of the SINK term are g $N/m^3/day$, a conversion factor is required. This is computed from Equation 164:

$$NH_3 + 2\ O_2 = HNO_3 + H_2O \qquad (164)$$

The oxygen consumption here is 4.57 g O_2/g N. Some authors prefer a value of 4.33 g O_2/g N, arguing that some of the ammonia consumed by the nitrifiers goes into cell synthesis (Garland 1978). However, the nitrifiers are themselves subject to predation, which returns the incorporated ammonia back to the stream for further nitrification. The actual amount of ammonia removed or nitrified is unknown, but the difference between 4.57 and 4.33 is insignificant when one considers the usual accuracy of field data. Therefore, the higher figure has been chosen, because it is somewhat conservative.

139. The rate of oxygen uptake resulting from stream reaeration can be formulated as:

$$\left\{\begin{array}{l}\text{Rate of reaeration}\\ \text{per unit volume}\end{array}\right\} = K2*(DOSAT-DO) \qquad (165)$$

where

K_2 = reaeration rate coefficient, day^{-1}

DOSAT = local solubility of oxygen, g O_2/m^3

DO = local oxygen concentration, g O_2/m^3

As discussed in Part V, all K2 equations may be rearranged to the following form:

$$K2 = AG*(U**E1)/(H**E2)*(1.024**(TEMP-20.)) \quad (166)$$

where

AG, E1, E2 = empirical coefficients

In using Equation 166, it must be noted that AG is sometimes given for 25° C. For those who prefer the Tsivoglou-Wallace (1972) equation in its published form, provision has been made in the program to permit its use as an option:

$$K2 = TSIV*(E(I-1)-E(I))/(DX/U)*1.024**(TEMP-20) \quad (167)$$

where

TSIV = empirical coefficient, m^{-1} (the value is entered in units of ft^{-1} and converted)

E(I-1), E(I) = water surface elevations at the upstream and downstream ends of the reach under consideration, m

DX = length of the reach, m

U = reach stream velocity, m/day

In using Equation 167, it must be noted that TSIV is sometimes given for 25° C. As discussed in Part V, the model can also account for structural and wind-driven reaeration.

140. The solubility of oxygen in water is calculated using the following formula (Elmore and Hayes 1960):

$$DOSAT = 14.652 + (-0.41022 + (0.007991 - 0.000077774*TEMP)*TEMP)*TEMP \quad (168)$$

141. The oxidation of reduced iron (FE) and manganese (Mn) is considered to occur as a first-order reaction, as discussed in paragraphs 147-149. Accounting for stoichiometric conversions, the oxygen used during oxidation of reduced iron and manganese is

$$\left\{\begin{array}{l}\text{Rate of oxygen used}\\ \text{for oxidation of Fe}\\ \text{and Mn, g } O_2/m^3/\text{day}\end{array}\right\} = -0.14*KFEDK*FE-0.15*KMNDK*MN \quad (169)$$

where

KFEDK = oxidation rate for iron, day^{-1}

FE = concentration of reduced iron, g/m^3

KMNDK = oxidation rate for manganese, day^{-1}

MN = concentration of reduced manganese, g/m^3

142. The complete balance of DO reactions is:

$$\left\{\begin{array}{l}\text{Rate of accumulation}\\ \text{of dissolved oxygen,}\\ \text{g } O_2/m^3/\text{day}\end{array}\right\} = K2*(DOSAT-DO)-K1*CBODNS-4.57*KN*NH3N$$
$$+(1.59+0.35*(NO3N/(NO3N+NH3N)))*ALGRO-1.59*ALGADK \quad (170)$$
$$-0.14*KFEDK*FE-0.15*KMNDK*MN$$

Phosphorus

143. The presumed sinks for phosphorus (P) are sorption to the sediments (Taylor and Kunishi 1971) and plant uptake. These two mechanisms appear to be competitive, and phosphate sorbed to sediments may be unavailable to plants (Fitzgerald 1970). Phosphate is released by algal decay.

144. The physicochemical sorption occurs only on aerobic sediments, and phosphate is released from anaerobic conditions (Gummerman 1970, Olsen 1964). This latter possibility is ignored in the present model under the assumption that anaerobic conditions are uncommon in streams. As much as half the phosphate present in streams subjected to detergent phosphate discharges may be polymerized as either tripolyphosphate or pyrophosphate (Engelbrecht and Morgan 1959). These different forms are believed to be sorbed and consumed at the same rates, so they have not been distinguished in the present model.

145. The phosphate sorption process is represented as a simple first-order decay:

$$\left\{\begin{array}{l}\text{Rate of sorption}\\ \text{of phosphate}\end{array}\right\} = -KPO4DK*PO4 \quad (171)$$

where

KPO4DK = first-order reaction rate coefficient, day^{-1}

PO4 = local phosphate concentration, g P/m^3

The rate coefficient is obtained from

$$KPO4DK = (APO4*Q**BK)*(1.024**(TEMP-20.)) \quad (172)$$

where

APO4 = empirical coefficient reflecting the adsorption of phosphate, day^{-1} $(m^3/day)^{-BK}$; if the turbulence option is not used, then the units are day^{-1} and Q**BK equals 1.0

146. The algal sources and sinks both are evaluated using Equations 160 and 161. These equations indicate that algae are about 1.0 percent P by weight. Therefore, the plant source and sink terms are:

$$\left\{\begin{array}{l}\text{Release of P by} \\ \text{algal decay}\end{array}\right\} = 0.01*ALGADK \quad (173)$$

$$\left\{\begin{array}{l}\text{Uptake of P by} \\ \text{algal growth}\end{array}\right\} = -0.01*ALGRO \quad (174)$$

The complete phosphate balance is:

$$\left\{\begin{array}{l}\text{Rate of accumulation of} \\ PO_4\text{-P per unit volume,} \\ \text{g } P/m^3/day\end{array}\right\} = -KPO4DK*PO4+0.01*(ALGADK-ALGRO) \quad (175)$$

Iron and Manganese

147. As modeled in CE-QUAL-RIV1, iron and manganese have no autochthonous sources. They can enter the system only in reduced form from upstream boundaries, tributaries, and lateral inflows. Oxygen depletion resulting from the oxidation of these reduced metals may have an adverse impact on water quality downstream from the inflow source. Since the presence of reduced metals probably accompanies releases of waters already somewhat depleted of DO, metal oxidation may exacerbate existing conditions.

148. Oxidation of reduced iron or manganese (loss from the system) is treated simply as a first-order process.

$$\begin{Bmatrix}\text{Loss of Mn, g/m}^3\text{/day} \\ \text{due to oxidation}\end{Bmatrix} = -\text{KMNDK*MN}$$

$$\begin{Bmatrix}\text{Loss of Fe, g/m}^3\text{/day} \\ \text{due to oxidation}\end{Bmatrix} = -\text{KFEDK*FE} \qquad (176)$$

Oxidation does not occur if DO is less than the user-specified value of OXIDAT. Use of Equation 176 requires field data to estimate oxidation (i.e., decay) rates.

149. Temperature and turbulence do not affect the oxidation rates in the model. Such rate corrections are unnecessary since the current model formulation does not account for other, possibly more important effects such as pH and autocatalysis. Future code improvements may allow implementation of a more mechanistic approach, accounting for variations in process rates.

Coliform Bacteria and Miscellany

150. Fecal coliform bacteria, which are a preferred indicator for total coliform bacteria, enter the system only through inflows that represent agricultural or urban flows. Fecal coliforms do not reproduce in natural aquatic environments, and their populations decay exponentially; thus

$$\begin{Bmatrix}\text{Loss of fecal coliforms,} \\ \text{col/100 ml/day}\end{Bmatrix} = -\text{KCOLIDK*1.047**(TEMP-20.)*COLI} \qquad (177)$$

where

KCOLIDK = fecal coliform die-off rate, day^{-1}

COLI = fecal coliform count, col/100 ml

151. The coliform bacteria variable, C(10,I) , can be used for modeling other miscellaneous nonconservative or conservative constituents since it does not interact with other water quality variables. Equation 177 does not contribute to the SINKS term of the transport equation (Equation 76) and KCOLIDK is the K_s term of Equation 76. Therefore, the first-order loss of

other nonconservative constituents could be represented by Equation 177 and the variable C(10,I) . Likewise, a conservative constituent could be modeled by settling KCOLIDK = 0.0 .

PART VII: THE HYDRODYNAMIC PROGRAM (RIV1H)

152. As has been noted earlier, the model equations for constituent transport are dependent upon hydraulic variables, but not vice versa. Because of this uncoupling, two entirely separate computer programs can be written. The first program, called RIV1H, solves for the time and space distribution of flow, cross-section area, top width, and depth. When completed, these calculations are stored for use in RIV1Q, the water quality model. RIV1Q is structured so that for any one set of hydraulic data, a variety of water quality simulations can be performed. This part describes in detail the structure and operation of RIV1H.

Overview

153. The hydrodynamic program contains five subroutines in addition to the main program. The main program sets up the flexible dimensioning of arrays given the number of nodes and segments. Subroutine MAIN2 handles the remainder of the input and output. It sorts through the tributary network, calling subroutine BUBBLE, and determines what types of boundary conditions are to be found. It sets up the main time march in which values for all nodes are solved simultaneously while time-steps are handled successively. At each time-step, boundary conditions are assigned, and subroutine CALC is called for each segment in a downstream order (from lowest order streams to higher order). The first iteration is completed by calling subroutine NEW for each segment in an upstream order. Subsequent iterations begin with calls to ITER, an entry point within CALC, and again are completed by calls to NEW. The iteration continues until values of the controlling variables, flow and area, have converged to within tolerance or up to a limit of 50 iterations. Subroutine CALC contains the governing equations--conservation of mass and momentum. Using the residuals and partial derivatives from these equations, it sets up a multidimensional Newton-Raphson iteration in matrix format. The matrix equation is reduced to two vectors and solved completely by subroutine MAT5. Subroutine NEW updates the values of flow, area, and other hydrodynamic variables. In the case of tributaries, it also completes the solution of the matrix equation.

Structure of RIV1H

154. Although the hydrodynamic code is written primarily in FORTRAN IV, the program takes advantage of several non-ANSI (American National Standard Institute) features of the FORTRAN compiler, such as namelist and list-directed input and output, alternative entry points into a subroutine, etc. All non-ANSI features of this program and RIV1Q are covered in Appendix A. The remainder of this section is a guided tour; that is, topics are discussed in the order they appear in the listing, which is found in Appendix B. Certain topics, such as the tributary structure, boundary conditions, cross-section formulas, and input and output are mentioned briefly in this section but are covered much more fully in their individual sections. The meaning and use of all the variable names are given in Appendix D, but it should be noted that in order to conserve storage, several arrays are reused in different ways in different sections of the program. The flowchart for RIV1H appears in Figure 12.

Main program

155. The main program reads the title card and then the grid card (see Input and Output, paragraphs 265-287) in which the user indicates the number of nodes, time-steps, and segments in the system. For the purposes of this program, a segment is defined as a stretch of stream, each boundary of which is either a system boundary, a receiving stream, or a control structure (see Figure 13). Forty arrays are assigned locations within the main storage array, DJ . The total storage required is given by

$$30*MNODE + 32*NS - 6 \qquad (178)$$

where MNODE is the number of nodes and NS is the number of segments in the system. The dimensions of these arrays are passed to subroutine MAIN2 by being placed in COMMON along with the title. Information on the number of time-steps is not used by RIV1H, as the program automatically stops upon exhaustion of the input data. The water quality, however, because of its facility for sensitivity analysis, requires this information and receives it from the hydrodynamic model. Once the main program calls MAIN2, its task is completed. Upon return to the main program, execution terminates.

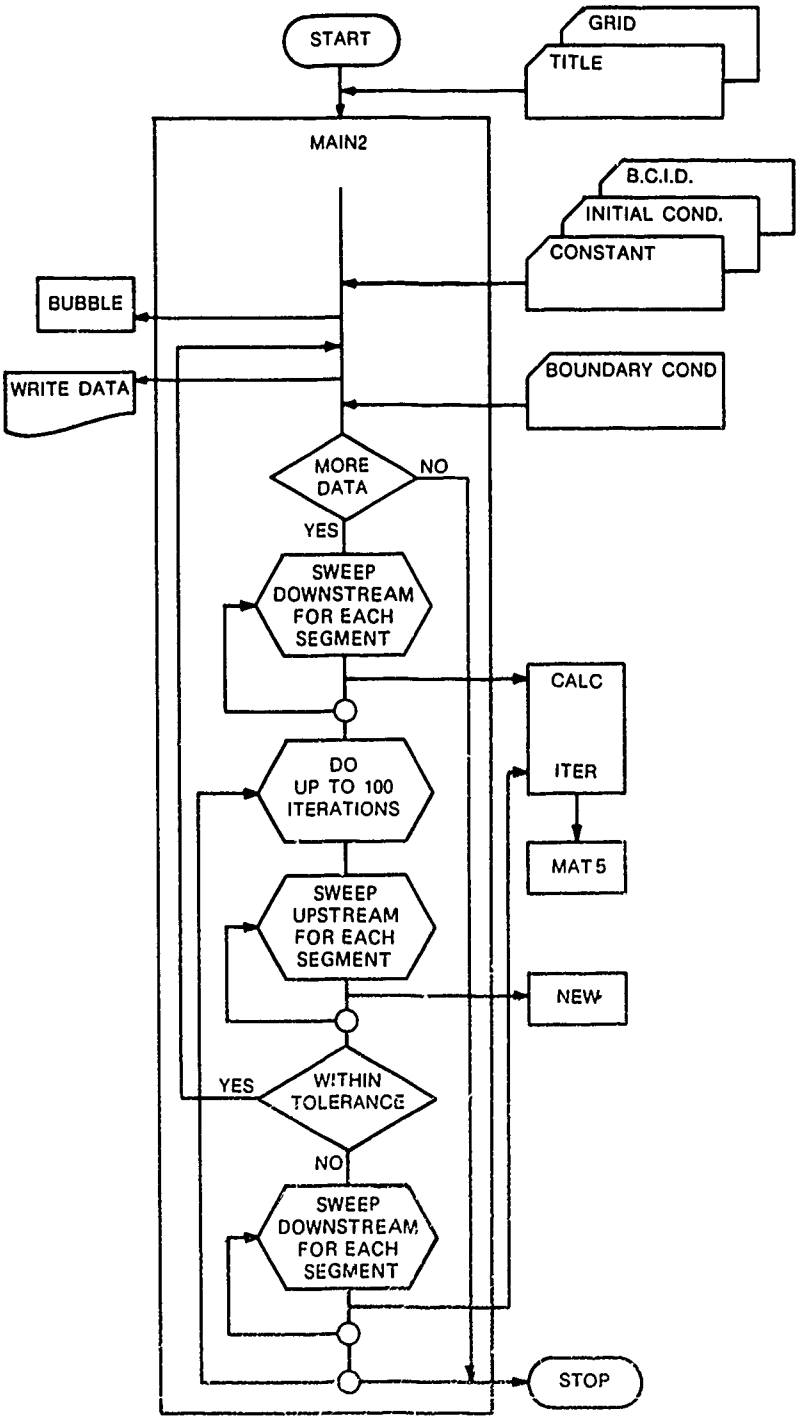


Figure 12. RIVlH organizational flowchart

SEGMENT NUMBER AND ARRAY POSITION	NNODE	FEEDS	JNODE	COSP	ORDER	IT0	IT1	IT2
1	18	3	10	60	7	2	0	0
2	23	3	17	60	3	3	0	0
3	26	7	0		6	0	2	4
4	26	3	21	60	1	4	6	6
5	13	4	17	30	2	6	0	0
6	25	7	8	60	4	1	5	5
7	16	0	0		8	0	1	1
8	9	6	10	120	5	5	0	0

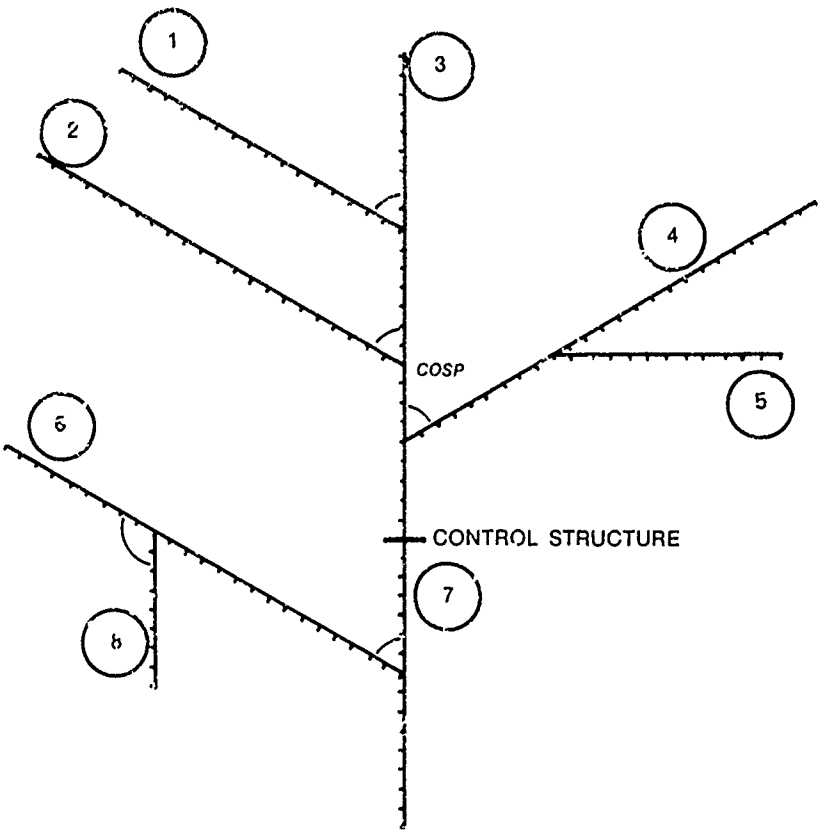


Figure 13. Sample network

Subroutine MAIN2

156. The subroutine MAIN2 is the executive block of RIV1H. It assigns default values for the constants GR, THETA, TOLER, BETA, FLOW, RCURVE, and IPRINT. The acceleration caused by gravity, GR, is set at 32.174 ft/sec^2 under the assumption that the units used in modeling are the customary English units. If the user wishes to use SI units, then a value of 9.80 m/sec^2 should be input (see Input and Output, paragraphs 265-287). In this case, Manning's coefficients should be adjusted by a factor of 1.49 to account for the SI form of the Manning's equation. However, the water quality program does not accept input data with SI units, although it converts to SI units.

157. The weighting factor, THETA, is discussed in Part III. A default value of 0.55 is cited in the literature as optimal for model accuracy; however, a higher value (i.e., 0.6 to 0.75) is often used to enhance stability. The relative error tolerance, TOLER, is assigned a default value of 0.001. Iteration ceases when all residuals (i.e., differences in successive iterations) for flow and area are less than TOLER times the root mean square of all flows or areas in the system. Experience suggests a larger value (i.e., 0.1) can reduce run time without substantially sacrificing accuracy. The momentum correction factor, BETA, is given the default value of 1.0, that is, no correction. The variables FLOW and RCURVE are used in input operations (see Input and Output, paragraphs 265-287). IPRINT is the print interval. Its default value is 1; that is, every time-step is printed.

158. The program then proceeds to input operations. The namelist dictionary is given, and the constant card is read. For each segment, the segment card is read and parsed; then initial conditions cards are read, one for each node. Following these is the boundary conditions identification card, in which the user informs the program which boundary conditions are being supplied in the input stream.

159. The user refers to each segment by its identification number, but the program refers to each segment by its order of appearance in the input. All references by identification number are therefore converted to references by array index, using the array ID as a dictionary. In the same way, it is convenient for the user to know which segment each boundary condition belongs to, and the program stores this information in the array IBC. For the program however, it is more convenient to know at which array location a particular boundary condition is to be found. The appropriate indices are determined and

stored in the arrays JBCU and JBCD, for upstream and downstream respectively. Next, the number of boundary condition updates, the update intervals, and the boundary conditions are read.

160. As mentioned previously, the iteration process consists of one downstream and one upstream sweep through the system. The next block of the program determines the proper ordering of the segments for the two sweeps and establishes the packing of arrays through which tributaries and receiving streams will pass information. The water quality model requires that data for all tributaries to any given segment be packed in a downstream order. This sorting is done by subroutine BUBBLE. Also in the program section, boundary conditions that are generated within the stream network, as opposed to being given in the input, are reset in the arrays JBCU and JBCD.

161. For the user, the controlling variables are flow and stage, but the program works predominantly with flow and area. The next section calculates area and channel top-width from the initial stage and the cross section given for each (see Cross-Section Formulas, paragraphs 181-185).

162. River miles are calculated for the system nodes as follows. Proceeding upstream through the network, if a segment is the mainstream of the system, the river mile of the last node is set to RMILEO (which is obtained from the input). If it is a tributary, the river mile of the last node is zero. If it terminates at a control structure, then it is considered to be part of the same stream as the segment downstream, and the river mile of the last node is equal to that of the latter's first node. Once the river mile of the most downstream node is established, the river miles of the rest of the nodes of the segment are calculated by successive addition of distance increments, indexing upstream.

163. The acceleration caused by gravity is divided by two in most places that it appears in the governing equations, so to streamline the program, this division is carried out once and for all before the time march begins.

164. A purely relative tolerance test risks a zero divide error in the case of a flow reversal where at some instant flow may go to zero. For this reason and others, the difference between successive estimates of the dependent variables flow and area is compared not with the values themselves but with the root-mean-square values for all nodes, one for flow and the other for area, times TOLER.

165. The input title, constants, and initial conditions are copied onto output. Grid parameters, network organization variables, distance increments, lateral inflows, and river miles are written onto disk storage for use by the water quality model.

166. The main time march begins by initializing the time-step number (ITIME) and the elapsed time (ELAPSE) to zero. Data from the previous time-step are then written (in the case of the first time-step, this refers to initial conditions). At the top of the page comes the program title. The run title, supplied by the user, is written below. Next the elapsed time, which has units of seconds for input and computation, is written out in hours and minutes; next to this appear the time-step number, the segment identification number, the segment, and name. Headings are written for river mile, flow, area, width, stage, and water surface elevation. Data appear in columns below the headings. Flow, area, width, lateral inflow, and water surface elevation are written onto disk storage. After incrementing the time-step number and elapsed time, the calculation for the next time-step proceeds.

167. One complete iteration of the Newton-Raphson method consists of a downstream sweep through the system, an upstream sweep, and a convergence test. For the first iteration within a time-step, the downstream sweep consists of updating all time-varying boundary conditions and calling subroutine CALC for each segment. Subsequent iterations require updating only those boundary conditions that are internally generated (i.e., rating curves and water surface elevation at tributary junctions). In addition, certain calculations within CALC are bypassed by entering the subroutine at entry ITER. The upstream sweep in both cases calls subroutine NEW for each segment.

168. Fifty iterations are permitted. If results have not converged to within tolerance by then, the program stops, writes an error message, and puts out a condition code of 7. The condition code (which has a value of zero if instead the run is successful) can serve as a signal to other job steps not to run.

Subroutine CALC

169. Subroutine CALC contains the governing equations of the hydrodynamic model--conservation of mass and conservation of momentum. It sets up these equations as the sum of terms having to do with the previous time-step and terms having to do with the current time-step. Terms having to do with the previous time-step need be calculated only once per time-step. Therefore,

all iterations past the first enter subroutine CALC at entry ITER, after the calculation of previous time-step terms. The terms are lumped into the array XC for the continuity or conservation of mass equation and XM for the conservation of momentum equation. The final entries of the arrays XC and XM are reserved for the contribution of the segment to the continuity and momentum equations respectively of its receiving stream (see Tributaries, paragraphs 189-208).

170. To enhance execution, repeatedly used array quantities are assigned to scalar variables; for instance, when dealing with the reach between nodes i and i+1 , the value of Q(I) is assigned to Q0 , the value of Q(I+1) is assigned to Q1 , and DX takes on the value DX1(I) .

171. The governing equations are written in the program as they appear in Part III with two exceptions: the off-channel storage area, A0 , is ignored and the slope, S0 , is combined with the pressure gradient term and appears in the equation as

$$DE = EL(I+1) - EL(I) \tag{179}$$

where EL is the water surface elevation.

172. The upstream boundary condition (see Boundary Conditions, paragraphs 186-188) can be either a flow or a stage. If a stage is used, it must be converted to the cross-sectional area by the appropriate formula (see Cross-Section Formulas, paragraphs 181-185). The set of boundary conditions employed is indicated by the value of LIB (see Boundary Conditions, paragraphs 186-188). The partial derivative array, AA, and the residual vector, R, are packed accordingly. Downstream boundary conditions can be flow, stage, or a rating curve and are handled in the same way.

173. The governing equations are given next, but a little preparatory work is necessary first. In calculating the partial derivatives of the governing equations with respect to A_i and A_{i+1} (cross-sectional areas of nodes i and i+1 , respectively), values for dB_i/dA_i and dB_{i+1}/dA_{i+1} are required. Applying the chain rule:

$$\frac{dB}{dA} = \frac{dB}{dH} \cdot \frac{dH}{dA} = \frac{dB}{dH} \cdot \frac{1}{B} \tag{180}$$

It remains to calculate dB_i/dH_i and dB_{i+1}/dH_{i+1}. Now as calculation proceeds from node to node, dB_{i+1}/dH_{i+1} is updated as the variable, DBDH. It takes on its new value in between the calculation of the partial of the momentum equation with respect to A_i and the partial with respect to A_{i+1}. This leaves the values of dA_1/dH_1, which must then be calculated, outside of the loop.

174. Proceeding in order then, the reach specific quantities are set. The residual of the continuity equation is calculated. The residual of the momentum is calculated. Calling the continuity equation F and the momentum equation G, $\partial F/\partial Q_i$, $\partial F/\partial A_i$, $\partial F/\partial Q_{i+1}$, $\partial F/\partial A_{i+1}$, $\partial G/\partial Q_i$, $\partial G/\partial A_i$, and $\partial G/\partial Q_{i+1}$ are calculated. A new value is calculated for DBDH to permit calculation of $\partial G/\partial A_{i+1}$. The matrix of partial derivatives, AA, is packed appropriately (see Boundary Conditions, paragraphs 186-188).

175. The residuals and partial derivatives are adjusted to account for the effects of any tributaries entering this segment. Next, allowance is made for the fact that the matrix solver requires nonzero entries in the upper left-hand and lower right-hand corners of the coefficient matrix, AA. Under certain types of boundary conditions, i.e. those involving stage at the upstream end or flow at the downstream end, this assumption would be invalid. To compensate, the program interchanges columns in the coefficient matrix prior to calling the matrix solver, subroutine MAT5, and reverses the interchange in the solution vector upon return.

176. Finally, before returning to MAIN2, if this segment is a tributary, it loads data into the tributary information passing array T.

Subroutine MAT5

177. Subroutine MAT5 contains an algorithm for solving the five-banded coefficient matrix AA constructed in subroutine CALC. It takes advantage of several facts about this matrix to increase efficiency. The matrix's structure is shown in Equation 195. The value of unity in the upper left-hand corner is implied, and the extreme diagonals are only half-populated. By pairwise row operations, these extreme diagonals are eliminated, leaving a matrix of tridiagonal structure. This is readily solved by a recursive Gaussian elimination (Carnahan, Luther, and Wilkes 1969) known as the Thomas algorithm. First, the lower codiagonal is eliminated proceeding from the top of the matrix down. Proceeding from the bottom up, the upper codiagonal is eliminated, and each row is normalized to leave the identity matrix on the

left and the solution vector on the right. In the case of a tributary, a companion vecto , C , is generated in this last step (see Tributaries, paragraphs 189-20 , Normally, the lower right-hand element is assumed to be unity, and . ement to its left is assumed to be zero. These assumptions would not be ase for a segment whose downstream boundary is a rating curve. Such a se nent must have its last row handled in a special way, which produces modified values for the solution vector (but not the companion vector, C , as such a segment cannot be a tributary).

Subroutine NEW

178. The output for subroutine MAT5 is vector R, which now contains adjustments to be made in flow and area to arrive at the new estimates. First, however, the vector R may have to be adjusted to account for effects of the receiving stream if this segment is a tributary. Once this is done, alternate values of R are added alternatively to flow and area. From the new area estimates, new stage and width estimates can be made. Now the cross-section formulas explicitly yield area and width, given stage. There is no explicit formulation, however, that will yield width and stage given area. The answer is therefore arrived at using another Newton iterative procedure. This is of the form

$$H_{new} = H_{old} - \frac{f\left(H_{old}\right) - A}{f'\left(H_{old}\right)} \qquad (181)$$

where

H_{new} = new estimate of stage
H_{old} = previous estimate of stage
$f(H)$ = cross-section formula for area as a function of stage
A = new cross-sectional area
$f'(H)$ = derivative of the cross-section formula for area as a function of stage with respect to stage

The variable $f'(H)$ is equivalent to the cross-section formula for width as a function of stage, and so it is called BO in the program, that is, a provisional estimate of width just as HO is a provisional estimate of stage.

179. The test of satisfactory convergence is the relative difference between successive estimates, where the tolerance is fixed at 0.01. Failure

to converge in 10 iterations causes the program to terminate, print an error message, and put out a condition code of 7.

Subroutine BUBBLE

180. As mentioned previously, the water quality model requires the tributary-data-passing arrays, JT and T to be packed such that all tributaries entering a given segment can be accessed in downstream order. The subroutine scans the list of segments to see if any has more than one tributary. If one is found, pointers to the tributaries and entries in the array JT are sorted in a downstream order according to the BUBBLE algorithm (Knuth 1973).

Cross-Section Formulas

181. It is possible to have a program of this sort work with explicit cross-section representations; that is, the coordinates measured in the field could be used directly to relate area, width, and depth. As short-hand cross-section descriptions, however, RIV1H has two stage-area and stage-width equations (stage is height of the water surface above the channel bottom): a hybrid power function and an ellipsoid function.

182. The hybrid power function (standard formula) is:

$$A = C_1 H + C_2 H^{C_3} \tag{182}$$

Consequently, since B is equal to dA/dH :

$$B = C_1 + C_2 C_3 H^{C_3 - 1} \tag{183}$$

where

A = area

H = stage

B = channel top width

These formulas can describe a variety of standard shapes. With $C_2 = 0$, they describe a rectangle of width C_1 . With $C_1 = 0$ and $C_3 = 2$, they describe a triangle with a height-to-width ratio of $\frac{1}{2C_2}$. With $C_3 = 2$

and $C_2 = \frac{1}{2}\left(\frac{1}{Y} + \frac{1}{Z}\right)$, they describe a trapezoid of bottom-width C_1 and side-slopes of Y and Z. With $C_1 = 0$, $C_2 = \frac{4}{3}\alpha^{1/2}$, and $C_3 = \frac{3}{2}$, they describe a parabolic cross section of the form

$$H = \frac{1}{\alpha}\left(\frac{B}{2}\right)^2 \tag{184}$$

where α is the distance from the vertex to the focus. Other shapes can be approximated by curve fitting.

183. In cases where an ellipse would give a better fit (for instance, flow in a partially full conduit), the user has the option of so describing such cross sections. The geometric descriptions at each node are independent; for instance, ellipses may be interspersed with standard descriptions. In designating the ellipse, C_1 is half the vertical axis length, C_2 is half the horizontal axis length, and C_3 is set to -1 to indicate to the program that an ellipsoid description is intended. If $C_1 = C_2$, of course, the cross section is circular. The formulas are

$$B = 2\frac{C_2}{C_1}\sqrt{2C_1H - H^2} \tag{185}$$

and

$$A = C_1C_2 \arccos(1-H/C_1) - B(C_1-H)/2. \tag{186}$$

All formulas are summarized in Figure 14.

184. In the course of the program, several manipulations of the cross-section formulas are required. These formulas (Equations 185 and 186) are found in subroutine MAIN2 in the initial calculations of areas and widths, again in subroutine CALC in performing the same calculations for upstream and downstream boundary conditions, and in subroutine NEW where the stages are updated. Later in subroutine CALC, when calculating the partial derivatives of the momentum equation with respect to area, a value is required for dB/dA. This is computed using Equation 183; it is equal to $\frac{1}{B}\frac{dB}{dH}$. The

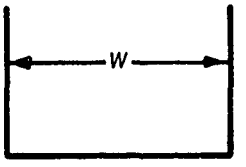
CROSS-SECTIONS FALL INTO TWO BASIC TYPES:

1. THE STANDARD FORMULA:

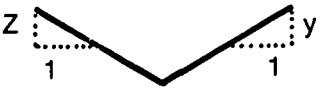
$A = C_1H + C_2H^{C_3}$, $B = C_1 + C_2C_3H^{C_3-1}$

IN THIS CATEGORY FALLS

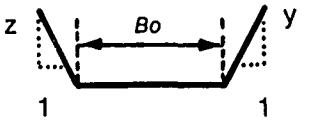
RECTANGULAR,
$C_1 = W$
$C_1 = O$



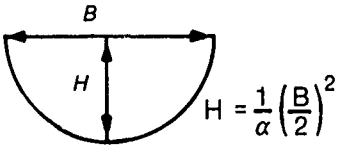
TRIANGULAR,
$C_1 = O$
$C_2 = \frac{1}{2}\left(\frac{1}{Z} + \frac{1}{y}\right)$
$C_3 = 2$



TRAPEZOIDAL,
$C_1 = B_O$
$C_2 = \frac{1}{2}\left(\frac{1}{Z} + \frac{1}{y}\right)$
$C_3 = 2$



AND PARABOLIC,
$C_1 = O$
$C_2 = \frac{4}{3}\alpha^{1/2}$
$C_3 = \frac{3}{2}$



OTHER SHAPES CAN BE APPROXIMATED BY CURVE-FITTING

2. ELLIPSOID

$B = 2C_2/C_1 \sqrt{2C_1H-H^2}$

$A = C_1C_2\text{ARCCOS}(1-H/C_1) - B(C_1-H)/2$

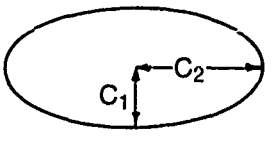


Figure 14. Cross-section types

derivative $\frac{dB}{dH}$ is called DBDH in the program and for the standard formula is equal to

$$C_2C_3\ (C_3 - 1)H^{(C_3-2)} \tag{187}$$

and for the ellipsoid formula

$$4\left(\frac{C_2}{C_1}\right)^2 \left(\frac{C_1 - H}{B}\right) \tag{188}$$

185. In most computer systems, algorithms for taking integral powers are much less involved than those for taking nonintegral powers. RIV1H takes advantage of this by determining for each cross section whether the exponent C3 is integral. If it is, the integer array I3 receives the value of C3, and C3 is set to zero, which serves as a signal to the rest of the program to use I3 in cross-section calculations rather than C3.

Boundary Conditions

186. For the system of equations to be solved, a pair of boundary conditions, one upstream and one downstream, must be known for all time-steps. Either flow or stage is given for the upstream condition, and any of flow, stage, or a rating curve can be given for the downstream condition. The six alternative sets are given in Table 3 along with values of the residual matrix (see Equation 195). The rating curve must be of the form

$$H = COEF\ Q^{EXPO} \tag{189}$$

where

H = depth of flow, ft

Q = discharge, ft^3/sec

Then COEF and EXPO are input in the "segment" card (see Input and Output, paragraphs 265-287). If the rating curve is instead of the form $Q = aH^b$, then simply set EXPO = 1/b and COEF = $a^{(-1/b)}$.

Table 3

Library (LIB) of Boundary Conditions

LIB	Upstream Condition	Downstream Condition	R_n	a_1	a_2	a_m	a_{m-1}
1	H	H	0	0	1	0	1
2	H	Q	0	0	1	1	0
3	H	RC	RC	0	1	DBCQ	DBCA
4	Q	H	0	1	0	0	1
5	Q	Q	0	1	0	1	0
6	Q	RC	RC	1	0	DBCQ	DBCA

Note: H = stage
Q = flow
RC = Equation 192
DBCQ = Equation 193
DBCA = Equation 194
R_n, a_1, a_2, a_m, a_{m-1} are stated in Equation 195

187. One purpose of the boundary conditions is to supply values for use in the next time-step. A boundary condition of flow can be used directly. A boundary condition of stage can be used after the area and width are calculated from it. The major purpose of the boundary conditions is to properly constrain the system of equations. This purpose is accomplished by treating the equations that assign values to the boundary conditions--or alternatively the rating curve equation--in the same way as the governing equations. The residual is placed in the residual matrix, R , and the partial derivatives are placed in the partial derivative matrix, AA . For instance, the assignment of the upstream boundary condition to H is H = BCU or A = f(BCU) where f is the stage-area function and BCU is the value of the upstream boundary condition. Isolating all known quantities on the left, we have

$$A - f(BCU) = 0 \tag{190}$$

The residual is equal to zero. The partial derivative with respect to A is 1. The partial derivative with respect to Q is 0. The same reasoning applies to the downstream condition and to flow, Q , as a boundary condition.

188. A slightly different situation is presented by having a rating curve as the downstream condition. Here the equation is

$$H = COEF*Q^{EXPO} \tag{191}$$

Now this is not an assignment statement, but a constraint on the behavior of H and Q . Since the solution scheme involves iterative approximation, Equation 191, when coupled with the governing equations, is not solved exactly. It leaves a non-zero residual.

$$H - COEF*Q^{EXPO} = R \tag{192}$$

This residual finds its place in the residual matrix, R. The partial derivatives of the above equation are taken with respect to Q and A . These are called in the program respectively DBCQ and DBCA,

$$\frac{\partial(-R)}{\partial Q} = EXPO*COEF*Q^{EXPO-1} \tag{193}$$

$$\frac{\partial(-R)}{\partial A} = -\frac{dH}{dA} = -\frac{1}{B} \tag{194}$$

The matrix becomes

$$\begin{bmatrix} a_1 & a_2 & & & & & & \\ DFQ & DFA & DFQ1 & DFA1 & & & & \\ DGQ & DGA & DGQ1 & DGA1 & 0 & & & \\ & 0 & DFQ & DFA & DFQ1 & DFA1 & & \\ & & DGQ & DGA & DGQ1 & DGA1 & 0 & \\ & & & 0 & DFQ & DFA & DFQ1 & DFA1 \\ & & & & DGQ & DGA & DGQ1 & DGA1 \\ & & & & & & a_{m-1} & a_m \end{bmatrix} \cdot \begin{bmatrix} \Delta Q_1 \\ \Delta A_1 \\ \Delta Q_2 \\ \Delta A_2 \\ \Delta Q_3 \\ \Delta A_3 \\ \Delta Q_4 \\ \Delta A_4 \end{bmatrix} = \begin{bmatrix} 0 \\ R1_1 \\ R2_1 \\ R1_2 \\ R2_2 \\ R1_3 \\ R2_3 \\ R_n \end{bmatrix} \tag{195}$$

with a_1 , a_2 , a_{m-1} , a_m , and R_n taking on values which depend on the value of LIB (see Table 3).

Tributaries

189. The major concerns in designing a program to handle tributaries are that: (a) the relevant physics be modeled with fidelity; (b) the program should be capable of handling any arbitrary tributary network; (c) it should be efficient in input and output, computer storage, and execution; and (d) it should be amenable to possible future change.

190. In this program, a stream network is envisioned as a group of discrete segments. A segment is defined as a stretch of stream whose boundaries are a system boundary, a receiving stream, or a control structure. No control structures are permitted within a segment. Based on user-supplied information about each segment, the program constructs the network, determines an upstream ordering of the segments, decides what types of boundary conditions apply to each segment boundary, and determines at which array locations each of these boundary conditions values is to be found. An upstream ordering assures that when the turn comes for any given segment to be processed, the segment it feeds into has already been processed. Conversely, a downstream ordering assures that when the turn comes for any given segment to be processed, all segments flowing into it have already been processed.

191. The first iteration of the Newton-Raphson method at each time-step requires new boundary conditions to be assigned at all time-varying boundaries. Subsequent iterations require new values only for boundary conditions that are generated within the systems, i.e., tributary junctions and flow through control structures. The program then assembles and solves the Newton-Raphson matrix for the entire system in what might be regarded as one step. In fact, however, the matrix is assembled and reduced segment by segment as the algorithm proceeds downstream. When the final segment is reached, the process is reversed, and the reduced, partitioned matrix is solved completely, segment by segment in an upstream order.

192. The detailed features of this algorithm can be illustrated by following the solution of a simple tributary system from the inside of the solution scheme out. The tributary, J , joins with the mainstream, M , at node K . In the remainder of this section, the portions of the governing equations relevant to tributary junctions are reviewed and placed within the matrix structure, and the solution of the matrix is delineated. Next, the way the main program controls the flow of information between subroutines and

coordinates the matrix assembly and solution is described. Finally, the method of constructing the network and assigning priority and boundary conditions is discussed.

The coefficient matrix

193 The governing equations are discussed in Parts II and III but are reviewed ... as the structure of the matrix solution is intimately tied to the structure of the equations. The Newton-Raphson method assembles a set of linear equations of the form

$$A \cdot x = R \tag{196}$$

where

A = coefficient matrix with elements of the form $\partial R_1/\partial x_j$

x = vector of increments in the controlling variables (Q and A)

R = vector of the negative residuals from the governing equations (continuity and momentum)

In the case of a single segment, the five-banded coefficient matrix is assembled as in the form of Equation 197, with three fully populated bands and two extreme bands half-populated.

$$\begin{bmatrix}
1 & & & & & & & & \\
\partial F_1/\partial Q_1 & \partial F_1/\partial A_1 & \partial F_1/\partial Q_2 & \partial F_1/\partial A_2 & & & & & \\
\partial G_1/\partial Q_1 & \partial G_1/\partial A_1 & \partial G_1/\partial Q_2 & \partial G_1/\partial A_2 & 0 & & & & \\
 & 0 & \partial F_2/\partial Q_2 & \partial F_2/\partial A_2 & \partial F_2/\partial Q_3 & \partial F_2/\partial A_3 & & & \\
 & & \partial G_2/\partial Q_2 & \partial G_2/\partial A_2 & \partial G_2/\partial Q_3 & \partial G_2/\partial A_3 & 0 & & \\
 & & & 0 & \partial F_3/\partial Q_3 & \partial F_3/\partial A_3 & \partial F_3/\partial Q_4 & \partial F_3/\partial A_4 & \\
 & & & & \partial G_3/\partial Q_3 & \partial G_3/\partial A_3 & \partial G_3/\partial Q_4 & \partial G_3/\partial A_4 & \\
 & & & & & & & 1 &
\end{bmatrix} \cdot \begin{bmatrix} \Delta Q_1 \\ \Delta A_1 \\ \Delta Q_2 \\ \Delta A_2 \\ \Delta Q_3 \\ \Delta A_3 \\ \Delta Q_4 \\ \Delta A_4 \end{bmatrix} = \begin{bmatrix} 0 \\ -F_1 \\ -G_1 \\ -F_2 \\ -G_2 \\ -F_3 \\ -G_3 \\ 0 \end{bmatrix} \tag{197}$$

194. In the case of a single tributary, two of these five-banded structures are placed head-to-toe and nine off-diagonal members are added to handle junction conditions, as in Equation 199.

195. The single off-diagonal element in the first quadrant $(-B_J/B_K)$ represents the condition that the water surface elevation at the mouth of the tributary is equal to that at the junction point on the mainstream, i.e. $E_J = E_K$. Taking differentials, $dE_J = dE_K$. However, the water surface elevation is the sum of the bed elevation and the depth of flow, and the bed elevation is taken to be a constant at any node, so $dH_J = dH_K$. To fit into the Newton-Raphson scheme, H must be replaced by the controlling variable A , or

$$dH_J = dA_J \cdot \frac{dH_J}{dA_J} = dA_J \; \frac{1}{B_J} = dA_K \, \frac{1}{B_K} \tag{198}$$

As it appears in the matrix

$$\left[\begin{array}{cccccccccccccccccc}
1 & & & & & & & & & & & & & & & & & \\
a_{11} & a_{12} & a_{13} & a_{14} & & & & & & & & & & & & & & \\
a_{21} & a_{22} & a_{23} & a_{24} & 0 & & & & & & & & & & & & & \\
 & 0 & a_{31} & a_{32} & a_{33} & a_{34} & & & & & & & & & & & & \\
 & & a_{41} & a_{42} & a_{43} & a_{44} & 0 & & & & & & & & & & & \\
 & & & 0 & a_{51} & a_{52} & a_{53} & a_{54} & & & & & & & & & & \\
 & & & & a_{61} & a_{62} & a_{63} & a_{64} & & & & & & & & & & \\
 & & & & & & & 1 & & & & & & -\frac{B_J}{B_K} & & & & \\
 & & & & & & & & 1 & & & & & & & & & \\
 & & & & & & & & a_{11} & a_{12} & a_{13} & a_{14} & & & & & & \\
 & & & & & & & & a_{21} & a_{22} & a_{23} & a_{24} & 0 & & & & & \\
 & & & & & & \frac{\partial F_{k-1}}{\partial Q_J} & \frac{\partial F_{k-1}}{\partial A_J} & & 0 & a_{31} & a_{32} & a_{33} & a_{34} & & & & \\
 & & & & & & \frac{\partial G_{k-1}}{\partial Q_J} & \frac{\partial G_{k-1}}{\partial A_J} & & & a_{41} & a_{42} & a_{43} & a_{44} & 0 & & & \\
 & & & & & & \frac{\partial F_{k}}{\partial Q_J} & \frac{\partial F_{k}}{\partial A_J} & & & & 0 & a_{51} & a_{52} & a_{53} & a_{54} & & \\
 & & & & & & \frac{\partial G_{k}}{\partial Q_J} & \frac{\partial G_{k}}{\partial A_J} & & & & & a_{61} & a_{62} & a_{63} & a_{64} & 0 & \\
 & & & & & & & & & & & & & 0 & a_{71} & a_{72} & a_{73} & a_{74} \\
 & & & & & & & & & & & & & & a_{81} & a_{82} & a_{83} & a_{84} \\
 & & & & & & & & & & & & & & & & & 1
\end{array}\right]
\left[\, X \,\right] = \left[\begin{array}{c} 0 \\ R_1 \\ R_2 \\ R_3 \\ R_4 \\ R_5 \\ R_6 \\ 0 \\ 0 \\ R_1 \\ R_2 \\ R_3 \\ R_4 \\ R_5 \\ R_6 \\ R_7 \\ R_8 \\ 0 \end{array}\right] \tag{199}$$

$$dA_J - dA_K \frac{B_J}{B_K} = 0$$

$$\left[1 - \frac{B_J}{B_K}\right] \begin{bmatrix} \Delta A_J \\ \\ \Delta A_K \end{bmatrix} = 0 \qquad (200)$$

196. The eight elements in third quadrant are $\frac{\partial F_{K-1}}{\partial Q_J}$, $\frac{\partial F_{K-1}}{\partial A_J}$, $\frac{\partial G_{K-1}}{\partial Q_J}$, $\frac{\partial G_{K-1}}{\partial A_J}$, $\frac{\partial F_K}{\partial Q_J}$, $\frac{\partial F_K}{\partial A_J}$, $\frac{\partial G_K}{\partial Q_J}$, $\frac{\partial G_K}{\partial A_J}$, where F and G are the residuals from the continuity equations, respectively, for the reaches above and below the junction point of the mainstream. These residuals must be adjusted for the presence of the tributary:

$$R_{2K-1} = -F_K + \frac{dt}{dX_K}\left[\theta Q_J + (1 - \theta)\, Q_J^j\right] = -F_K + \frac{F_J}{dX_K}$$

$$R_{2K} = -G_K + \frac{dt}{dX_K}\left[\theta \frac{Q_J^2}{A_J} + (1 - \theta) \frac{Q_J^{j2}}{A_J}\right] \cos\phi = -G_K + \frac{G_J}{dX_K} \qquad (201)$$

where

F, G = lumping of terms

j = previous time-step

ϕ = junction angle

The derivatives are

$$\frac{\partial F_K}{\partial Q_J} = -\frac{dt}{dX_K}\,\Theta$$

$$\frac{\partial F_K}{\partial A_J} = 0$$

$$\frac{\partial G_K}{\partial Q_J} = -\frac{2dt}{dX_K}\,\Theta\left(\frac{Q_J}{A_J}\right)\cos\,\phi$$

$$\frac{\partial G_K}{\partial A_J} = \frac{dt}{dX_K}\,\Theta\left(\frac{Q_J}{A_J}\right)^2\,\cos\,\phi \qquad (202)$$

197. Conceptually, the solution begins with reducing the second quadrant to the identity matrix by Gauss elimination. First, the extreme diagonals are eliminated by pair-wise operations on the rows. Next, the lower codiagonal is eliminated by standard tridiagonal matrix methods. Finally, in eliminating the upper codiagonal, the single first-quadrant element, which has to this point been untouched, generates a column of numbers above it (Equation 203). All elements of this column are proportional to the single original element in that column, $-B_J/B_K$. Because of this fact, that element can be given the value of $-B_J$ provisionally (as the tributary does not "know" the value of B_K), and the other elements can be multiplied by the factor $1/B_K$ later. The last and next-to-last rows in the tributary portion of the matrix can now be brought down to eliminate the eight (six non-zero) third-quadrant elements. This elimination changes the value of the four residuals around the junction node and the four elements of the coefficient matrix below the elements C_i. Following the numbering in Equations 199 and 203

$$
\begin{bmatrix}
1 & & & & & & & & & & & & & \\
 & 1 & & & & & & & & & & & & \\
 & & 1 & & & & & & & & & & & \\
 & & & 1 & & & & & & & & & & \\
 & & & & 1 & & & & & & & & & \\
 & & & & & 1 & & & & & & & & \\
 & & & & & & 1 & & & & & & & \\
 & & & & & & & 1 & & & & & & \\
 & & & & & & & & 1 & & & & & \\
 & & & & & & & & a_{11} & a_{12} & a_{13} & a_{14} & & \\
 & & & & & & & & a_{21} & a_{22} & a_{23} & a_{24} & & \\
 & & & & \frac{\partial F_{K-1}}{\partial Q_J} & 0 & & & a_{31} & a_{32} & a_{33} & a_{34} & & \\
 & & & & \frac{\partial G_{K-1}}{\partial Q_J} & \frac{\partial G_{K-1}}{\partial A_J} & & & a_{41} & a_{42} & a_{43} & a_{44} & & \\
 & & & & \frac{\partial F_K}{\partial Q_J} & 0 & & & & 0 & a_{51} & a_{52} & a_{53} & a_{54} \\
 & & & & \frac{\partial G_K}{\partial Q} & \frac{\partial G_K}{\partial A_J} & & & & & a_{61} & a_{62} & a_{63} & a_{64} \\
 & & & & & & & & & & & & a_{71} & a_{72} & a_{73} & a_{74} \\
 & & & & & & & & & & & & a_{81} & a_{82} & a_{83} & a_{84} \\
 & & & & & & & & & & & & & & & 1
\end{bmatrix}
\cdot
\begin{bmatrix} \\ \end{bmatrix}
\cdot X =
\begin{bmatrix}
0 \\
R_1^* \\
R_2^* \\
R_3^* \\
R_4^* \\
R_5^* \\
R_6^* \\
0 \\
0 \\
R_1 \\
R_2 \\
R_3 \\
R_4 \\
R_5 \\
R_6 \\
R_7 \\
R_8 \\
0
\end{bmatrix}
\qquad (203)
$$

$$a_{34} = \frac{\partial F_{K-1}}{\partial A_K} + \frac{dt}{dX_{K-1}} \Theta \frac{C_6}{B_K}$$

$$a_{44} = \frac{\partial G_{K-1}}{\partial A_K} + \frac{2dt}{dX_{K-1}} \cos \phi \Theta \frac{Q_J}{A_J} \frac{C_6}{B_K} + \frac{dt}{dX_{K-1}} \Theta \left(\frac{Q_J}{A_J}\right)^2 \frac{B_J}{B_K} \cos \phi$$

$$a_{52} = \frac{\partial F_K}{\partial A_k} + \frac{dt\Theta}{dX_K} \frac{C_6}{B_K}$$

$$a_{62} = \frac{\partial G_K}{\partial A_K} + \frac{2dt}{dX_K} \Theta \frac{Q_J}{A_J} \frac{C_6}{B_K} \cos \phi + \frac{dt}{dX_K} \Theta \left(\frac{Q_J}{A_J}\right)^2 \frac{B_J}{B_K} \cos \phi$$

$$R_3 = -F_{K-1} + \frac{F_J}{dX_{K-1}} + \frac{dt\Theta}{dX_{K-1}} R_6^*$$

$$R_4 = -G_{K-1} + \frac{G_J}{dX_{K-1}} + \frac{2dt}{dX_{K-1}} \frac{Q_J}{A_J} \cos \phi R_6^*$$

$$R_5 = -F_K + \frac{F_J}{dX_K} + \frac{dt\Theta}{dX} R_6^*$$

$$R_6 = -G_K + \frac{G_J}{dX_K} + \frac{2dt}{dX_K} \Theta \frac{Q_J}{A_J} \cos \phi R_6^* \qquad (204)$$

Terms that do not have to do with the main stem can be collected.

$$T_1 = dt \Theta C_6$$

$$T_2 = F_J + dt \Theta R_6^*$$

$$T_3 = 2dt \Theta \frac{Q_J}{A_J} C_6 + dt \Theta \left(\frac{Q_J}{A_J}\right)^2 B_J \cos \phi$$

$$T_4 = G_J + 2dt\ \Theta \frac{Q_J}{A_J}\ R_6^* \tag{205}$$

to leave

$$a_{39} = \frac{\partial F_{K-1}}{\partial A_K} + \frac{T_1}{dX_{K-1} B_K}$$

$$a_{44} = \frac{\partial G_{K-1}}{\partial A_K} + \frac{T_3}{dX_{K-1} B_K}$$

$$a_{52} = \frac{\partial F_K}{\partial A_K} + \frac{T_1}{dX_K B_K}$$

$$a_{62} = \frac{\partial G_K}{\partial A_K} + \frac{T_3}{dX_K B_K}$$

$$R_3 = -F_{K-1} + \frac{T_2}{dX_{K-1}}$$

$$R_4 = -G_{K-1} + \frac{T_4}{dX_{K-1}}$$

$$R_5 = -F_K + \frac{T_2}{dX_K}$$

$$R_6 = -G_{K-1} + \frac{T_4}{dX_K} \tag{206}$$

198. In the notation of the program, if the junction occurs at node i and IR = 2(i-1) and IA = 5*IR

$$AA(IA-6) = a_{34}$$

$$AA(IA-2) = a_{44}$$

$$AA(IA+2) = a_{52}$$

$$AA(IA+6) = a_{62}$$

$$R(IR) = R_3$$

$$R(IR+1) = R_4$$

$$R(IR+2) = R_5$$

$$R(IR+3) = R_6 \qquad (207)$$

The terms T_1 through T_4 are calculated in subroutine CALC when it solves the tributary portion of the matrix and are passed back to that subroutine when it is called to work on the main stem. The adjustment in the main stem portion of the matrix and residuals vector is made, and the main stem's system of equations can be solved completely by Gauss elimination. This leaves all main stem variables solved, but the tributary system as yet undetermined. Once the coefficient matrix A is known, however, the elements C_i can be eliminated from the tributary matrix and R_i^{**} can be calculated from:

$$R_i^{**} = R_i^{*} - C_i \frac{\Delta A_K}{B_K} \qquad (208)$$

Then R_i^{**} is used to make adjustments in the controlling variables Q and A in the normal way.

Flow of information

199. The preceding scheme can be generalized to handle any arbitrary system. The tributary passes to the receiving stream the values T_1 through T_4 and also K, the junction node. The receiving stream passes back to the tributary the value $\Delta A_K/B_K$, placing it into T_1. In the general case, T is

an array of dimension (4,NS) where NS is the number of segments in the network. Other arrays involved in this information-passing process are JT(NS), which contains the junction nodes; ITO(NS); IT1(NS); and IT2(NS). The array ITO(L) tells where in the array T segment L should place its tributary information (ITO(L) = 0 if L is not a tributary). The array IT1(L) gives the index within the array T of information from the first tributary entering segment L, and IT2 gives the index of the information from the last tributary. If no tributaries enter segment L, IT1 and IT2 are set to zero. See Figure 13 for an example.

200. The main program works downstream through the network, and for each segment L, it calls subroutine CALC with the arguments T, JT, ITO(L), IT1(L), and IT2(L). In subroutine CALC, segment L picks up tributary information from T between the indices IT1 and IT2. The matrix solver is called and upon return, segment L loads its own information into T at index ITO. The solution is completed by sweeping upstream through the system; for each segment, subroutine NEW is called with the same arguments. This time the segment picks up information from T at index ITO and distributes information to its tributaries at indices IT1 through IT2.

201. One sweep down and one sweep up determines the system. No segment-wise iteration is necessary. Other advantages of this procedure are that (a) it allows backwaters up tributaries; (b) the coefficient matrix is assembled and solved in pieces as required; and (c) since a banded-style solution is done, the number of operations performed is of order N , where N is the total number of nodes in the system. In additic , since the scheme is generalized, the number of "bookkeeping" operations is of order M , where M is the number of segments in the system.

Network structure

202. The user provides in the input, NS , the total number of segments and then for each segment L gives (among other data), FEEDS(L) and JNODE(L), that is, which stream, if any, this segment flows into and the node on the receiving stream at which the junction occurs. If the two segments are parts of the same stream, but separated by a control structure, the input field of JNODE is left blank and so is taken to be zero.

203. From these data, the program infers the network structure, establishes an upstream ordering of the segments, determines where boundary conditions that are passed from segment to segment are to be found, and

establishes the packing of the tributary-information-passing arrays JT and T. The latter is done by fixing the values of IT0, IT1, and IT2 for each segment.

204. The program begins with the assumption that the system has a single terminus. Only one segment, the last segment of the main stem, does not flow into another segment. The index of this stream is the first entry in the array ORDER. Next the array FEEDS is scanned to find streams that feed into this first stream, and they are successively entered into ORDER. Next, the program looks at the second entry of the array ORDER and scans FEEDS for streams that flow into it. The process continues until the array ORDER is completely filled. Whenever a match is found between a tributary J and its receiving stream M, the program discriminates between JNODE equal to zero and JNODE greater than zero. In the former case, the two form a single stream separated by a control structure. The segment M therefore will find its upstream boundary condition at the last node of segment J. The latter is the case of a true tributary. Segment J will find its downstream boundary condition at the junction node on M. Also a new slot, IT, is called for the tributary-information arrays JT and T. The tributary J will deposit its information in slot IT. The receiving stream M will withdraw information from slot IT. Using the definitions of IT0, IT1, and IT2 given above, IT0(J) is set to IT, IT1(M) is set to IT if it has not already been set, and IT2(M) is set to IT.

Boundary conditions
in the tributary structure

205. For each segment, the user specifies what types of boundary conditions will apply upstream and downstream. Flow or stage (designated Q or H) may be given upstream, and flow, stage, or a rating curve (designated Q, H , or R) may be given downstream. The default at both ends is stage. Based on this input, the program assigns a value of one through six to LIB for that segment (Table 3).

206. The network structure may, however, override this assignment. A segment downstream of a control structure must have an upstream boundary condition of flow. A segment that is a true tributary must have a downstream boundary condition of stage.

207. Once the types of boundary conditions are established, the question is where the values of the boundary conditions are to come from. The answer is given in the arrays JBCU and JBCD for upstream and downstream,

respectively. The default is that the boundary condition is time-invariant and is given on the "segment" card of the input deck (see Input and Output formats, paragraphs 265-287). In this case JBCU or JBCD is given a value of zero. The second possibility is that they are supplied in the "boundary conditions" cards. The user indicates to the program which boundary conditions will appear there by means of the "boundary conditions identification" card. The value of JBCU or JBCD is set to the position of that piece of data on the boundary conditions card. Finally, as indicated above, the boundary condition may come from within the system; for example, the downstream boundary condition for segment L might come from node J. In this case JBCD(L) is set to -J.

208. When it is time to disburse boundary conditions, again three cases arise. If JBCU or JBCD is equal to zero, the boundary condition is time-invariant, and no change is made. If JBCU or JBCD is positive, then the boundary condition is obtained from that position in the input stream. If JBCU or JBCD is negative, the boundary condition is obtained from the node number -JBCU or -JBCD. As mentioned previously, an internally generated upstream boundary condition must be a flow. An internally generated downstream boundary condition must be a stage. Stage at the junction point on the receiving stream, however, is not equal to stage at the mount of the tributary--it must first be corrected for the difference in bed elevation between the two.

Input and Output Formats

Input

209. Input to RIV1H consists of nine types of cards: (a) a title card, (b) a grid card, (c) a constant card, (d) segment cards, (e) initial conditions cards, (f) a boundary conditions identification card, (g) an update card, (h) update interval cards, and (i) boundary conditions cards. The details of the input cards are discussed in the following paragraphs, and an example is given in Figure 15. Output consists of the input data followed by the values of the hydrodynamic variables for every node of each segment (one segment per page) for each time-step print interval. In addition to printed output, there is output onto disk storage for later use by RIV1Q or plotting programs.

```
SAMPLE DATA SET - USERS MANUAL
 40, 520, 3
&CONST BETA=1.0, GR= 32.17, RMILEO=0.0,THETA =0.67 ,TOLER=0.10,
 IPRINT=192, &END
1 MAIN STEM ABOVE REREG DAM                16  2   Q        Q 400.00
4014.  500.  10.0   0.0 495.0 690.0  7.00   2.0  .050   0.0 0.625 0.025
5248.  500.  10.2   0.0 494.0 153.0  3.40   2.0  .050   0.0 0.600 0.010
3983.  500.  10.8   0.0 492.0  80.0  5.70   2.0  .050   0.0 0.000 0.000
4334.  623.  10.2   0.0 485.0 185.0  2.10   2.0  .050   0.0 0.000 0.000
4428.  750.  10.4   0.0 484.0 240.0  2.10   2.0  .050   0.0 0.000 0.000
4428.  750.  10.4   0.0 479.0 300.0  1.85   2.0  .050   0.0 0.000 0.000
4933.  750.  10.8   0.0 474.0 355.0  1.60   2.0  .050   0.0 0.000 0.000
3667.  750.  10.9   0.0 472.0 175.0  3.20   2.0  .050   0.0 0.000 0.000
4200.  750.  10.7   0.0 472.0 225.0  0.70   2.0  .050   0.0 0.000 0.000
4200.  750.  10.7   0.0 470.0 224.0  0.80   2.0  .050   0.0 0.000 0.000
4200.  750.  10.7   0.0 468.0 223.0  0.90   2.0  .050   0.0 0.000 0.000
4200.  750.  10.5   0.0 466.0 221.0  1.00   2.0  .050   0.0 0.500 0.010
4664.  750.  10.5   0.0 463.5 220.0  1.10   2.0  .050   0.0 0.000 0.000
4664.  750.  10.4   0.0 460.5 215.0  1.80   2.0  .050   0.0 0.000 0.000
4664.  750.  10.5   0.0 457.5 210.0  2.50   2.0  .050   0.0 0.000 0.000
4352.  750.  10.4   0.0 455.0 205.0  3.20   2.0  .050   0.0 0.000 0.000
2 MAIN STEM BELOW REREG DAM                18       Q 400.    H 15.00
4352.  750.  10.2   0.0 455.0 203.0  2.80   2.0  .050   0.0 0.550 0.011
4352.  750.  10.9   0.0 455.0 200.0  2.40   2.0  .050   0.0 0.000 0.000
4352.  750.  10.5   0.0 455.0 198.0  2.00   2.0  .050   0.0 0.000 0.000
3733.  750.  10.8   0.0 455.5 195.0  1.60   2.0  .050   0.0 0.000 0.000
3733.  750.  10.8   0.0 453.5 200.0  1.30   2.0  .050   0.0 0.000 0.000
3733.  750.  10.6   0.0 451.5 205.0  1.00   2.0  .050   0.0 0.000 0.000
4498.  750.  10.2   0.0 449.0 210.0  0.60   2.0  .050   0.0 0.000 0.000
4498.  750.  10.3   0.0 445.0 225.0  0.90   2.0  .050   0.0 0.000 0.000
4498.  750.  10.4   0.0 442.0 245.0  1.30   2.0  .050   0.0 0.000 0.000
4498.  750.  10.9   0.0 439.0 260.0  1.60   2.0  .050   0.0 0.000 0.000
5772.  750.  10.4   0.0 435.5 280.0  2.00   2.0  .050   0.0 0.000 0.000
5772.  750.  10.8   0.0 434.5 300.0  2.00   2.0  .050   0.0 0.000 0.000
5772.  750.  10.5   0.0 433.5 325.0  2.00   2.0  .050   0.0 0.000 0.000
5772.  750.  10.3   0.0 432.5 345.0  2.00   2.0  .050   0.0 0.000 0.000
5140.  750.  10.2   0.0 432.5 370.0  1.60   2.0  .050   0.0 0.000 0.000
5140.  750.  10.1   0.0 428.5 610.0  1.75   2.0  .050   0.0 0.000 0.000
5140.  750.  15.0   0.0 424.5 850.0  1.90   2.0  .050   0.0 0.000 0.000
4450.  750.  15.0   0.0 420.01100.0  2.10   2.0  .050   0.0 0.000 0.000
3 TRIBUTARY UPSTRM OF REREG                 6  1  4Q 250.    H        45.0
2564.  250.  10.5   0.0 488.5 177.0  3.00   2.0  0.04    0.0 0.500 0.010
2970.  250.  10.2   0.0 488.0  40.0  1.90   2.0  0.04    0.0 0.000 0.000
2970.  250.  10.5   0.0 484.8 143.0  2.10   2.0  0.04    0.0 0.000 0.000
2793.  250.  10.7   0.0 481.5 147.0  2.30   2.0  0.04    0.0 0.000 0.000
1977.  250.  10.7   0.0 482.5 212.0  1.80   2.0  0.05    0.0 0.000 0.000
100.0  250.  10.7   0.0 482.5 250.0  2.40   2.0  0.05    0.0 0.000 0.000
+1/
      23
      96      120      144      164      184      200      220
     240      260      280      300      320      340      350
     370      390      400      420      440      460      480
     500      520
   450.    500.
   450.    500.
```

Figure 15. Sample RIV1H input (Continued)

```
450.    500.
450.    600.
450.    600.
450.    600.
450.    600.
450.    700.
450.    700.
450.    800.
450.    800.
450.    700.
450.    700.
450.    700.
450.    700.
450.    600.
450.    500.
450.    400.
450.    400.
450.    400.
450.    400.
450.    400.
450.    400.
```

Figure 15. (Concluded)

210. Title card. The user has the full 80 columns of the title card to write any appropriate title. It will be printed, verbatim, at the top of every page.

211. Grid card. The grid card defines the total number of nodes, time-steps, and segments in the run. The three numbers can appear anywhere on the card and can be separated by a comma or blanks or both, but they must appear in the following order: nodes, time-steps, segments. Note also that these numbers correspond to grid lines, not intervals. For instance, if the study area extends from river mile 1.0 to river mile 0.0 with 0.1 mile reaches, the number of nodes is 11, not 10. The same applies to time-steps.

212. Constant card. There are six variables that may be assigned values on the constant card: RMILE0, THETA, TOLER, GR, IPRINT, and BETA. The variable RMILE0 is the river mile of the first node. The variable THETA is the weighting factor (see "The Governing Equation," paragraphs 55-58). Historically, models of this type have used a value of 0.55 or greater for stability reasons. This is the default RIV1H uses.

213. With some applications, a value of THETA higher than 0.55 (between 0.55 and 1.0) may be desirable to reduce parasitic oscillations (Liggett and

Cunge 1975). Parasitic oscillations, which are actually numerical dispersion or phase errors associated with short waves, can cause modeling problems when the parasitic oscillation is on the same order of magnitude as the depth (a negative depth can result). Values of THETA between 0.6 and 0.75 have been used to reduce these oscillations without significant loss of accuracy. However, accuracy does decrease as THETA is increased because of increased numerical dampening. The requirement for numerical stability is $0.5 < \theta < 1.0$.

214. The variable TOLER is the maximum acceptable relative tolerance used by the program to decide whether another iteration is necessary. The default value is 0.001; that is, iteration will stop when the difference between current and previous estimates of flow and area for every node are less than 0.001 times the root mean square of all flows and areas, respectively, throughout the system. If the user desires greater accuracy and has a stable system, a lower value of TOLER may be desired. Conversely, if the user requires less accuracy and swifter computation time, then a higher value may be desired, such as 0.1. This value has yielded acceptable accuracy. It should be pointed out that TOLER is a control on precision from one time-step to the next--not from the beginning of the run to the end.

215. The variable GR is acceleration caused by gravity. Its default value is 32.174 ft/sec^2. If the user wishes to make a run in SI units, then GR should be set at 9.80 m/sec^2. Several things to keep in mind in this case are that the river miles in the output are meaningless, the Manning's coefficient must be multiplied by 1.49, and the output is not usable in RIVIQ. The variable IPRINT is the print interval (number of time-steps between prints) for output file 6. The variable BETA is the momentum correction factor, with a default value of 1.0.

216. The form of the constant card is as follows. The first space is left blank. Columns 2 through 7 contain &CONST. Next are the names of each variable followed by an equals sign followed by the value the user wishes to assign. If the user is content with the default value, then that variable need not appear on the constant card. Assignments are separated by commas. At the end of the list, anywhere on the card, the expression &END is placed (see Figure 15).

217. Segment card. On the segment card, the user specifies segment-specific data: ID, the identification number; SNAME, the name; NNODE, the number of nodes in this segment; FEEDS, which segment it flows into, if any;

JNODE, the number of the junction node on the receiving stream; BTU, the type of boundary condition imposed at the upstream end; BCU, a value for that boundary condition if it is time-invariant; BTD, the type of boundary condition imposed on the downstream end; BCD, a value for that boundary condition if it is time-invariant; and COSP, the junction angle in degrees if the segment is a tributary. The format is

I2,10A4,3I3,2(A1,F8.0),F8.0.

The choice of identification number and the ordering of the segment cards are arbitrary. The only constraint is that identification numbers used here must be consistent with those used in the water quality model and they must be positive. The segment name can be up to 40 characters long. It has no use in the model other than labeling output. NNODE counts the total number of nodes in the segment, which is one more than the number of reaches. FEEDS refers to the receiving stream by its identification number. JNODE refers to the junction node on the receiving stream, starting with the first node of the stream equal to one. Referring to Figure 13, JNODE for Segment 1 would be 10. When two segments are actually part of the same stream but separated by a control structure, JNODE is left blank (control structures are not permitted within a segment, only at the boundaries). The boundary type H signifies stage, depth above bottom (and is the default), Q signifies flow, and R signifies a rating curve. A rating curve is not a permissible upstream boundary condition. The value of the boundary condition is in feet for stage and cubic feet per second for flow. COSP is the angle formed by the tributary junction, in degrees. See Figure 13 for the sense of this angle. In the case of a rating curve of the form

$$H = COEF*Q^{EXPO} \qquad (191 \text{ bis})$$

The exponent EXPO is given in place of BCD and COEF in place of COSP. There is no conflict as a tributary cannot be terminated by a rating curve. If instead the rating curve is of the form $Q = aH^{L}$, then set $EXPO = 1/b$ and $COEF = a^{(-1/b)}$.

218. Initial conditions card. Each segment card is followed by a set of initial conditions cards, one per node. Twelve pieces of data can appear

on each initial conditions card. Each number can occupy a field of six. Any right-justified number without a decimal point will be taken as having a decimal point at the end (e.g., 25 will be read as 25.0). Any value not specified will be taken as zero. These pieces of data are, in order: length of the reach downstream from the node in feet (DX1); flow in cubic feet per second (Q); stage-height of water above the channel bottom in feet (H); lateral inflow in cubic feet per second per foot (QL) (flow divided by DX1); streambed elevation relative to some datum in feet (Z, initially EL is used and moved into Z array), (C1), (C2), and (C3) constants in the cross-section formula (see Cross-Section Formulas, paragraphs 181-185); Manning's coefficient for the reach downstream of the node (CN1); constriction energy loss coefficient for the reach downstream of the node (KE1); intercept of the linear equation relating Manning's n to depth of flow in the stream (AX); and the slope (DNDH) of the linear equation relating Manning's n to depth. The term KE1 should be used only when there is a rapid constriction in the channel and only after it has been determined that refining the grid work in the area of the constriction is infeasible. Values for KE1 range from 0.0 to 0.5, with the higher value associated with abrupt constriction.

219. Manning's n at a cross-section can vary with flow conditions in many streams. High values may be more representative for shallow-depth conditions whereas lower values may be appropriate for deeper flow. For time-varying flow conditions, a variable Manning's coefficient may have to be adjusted as a function of depth during the simulation. Typically, shoal areas exhibit a variable n that can have a significant effect on computed stage; at low stage, n is usually larger than at high stage. If the user wishes to allow for variable Manning's coefficients, the values of AX and DNDH in the input data, which appear on the Initial Conditions card, should be set to the appropriate values. If the user chooses not to vary Manning's n , these values should be set to 0. The equation relating Manning's n to depth is XMAN = AX - DNDH*H where H is the value for depth at a particular node. During initiation, XMAN is set to the original value (CN1) at that node. As the program executes, XMAN is adjusted with relation to depth over time. If at some point during the execution, the value of XMAN becomes less than 0.01, XMAN is reset to 0.01. If this event occurs, a message is printed to the diagnostic file, FILE7. The first location in the code where the adjusted

Manning's values occur is several lines after statement 301 while the second location is several lines after statement 170.

220. Boundary conditions identification card. On the boundary conditions identification card, the user indicates which boundary conditions will be given on the boundary conditions cards. As an example, consider a system with two segments separated by a control structure where Segment 1 is the reach above the structure and Segment 2 is the reach below. The upstream and downstream boundary conditions will be specified for Segment 1, whereas the downstream boundary condition will be specified for Segment 2. The upstream boundary condition for Segment 2 (below the control structure) is determined internally by the program. The card would appear as

+1,-1,-2/

with the numbers being the ID numbers, the positive sign standing for upstream, and the negative for downstream. The list must be terminated by a slash.

221. Update card. On the update card, the user indicates the total number (NUM) of boundary condition updates. The format is I10. In the following cards, a total of NUM boundary condition update intervals and NUM boundary condition cards must be provided.

222. Update interval card. On the update interval card, the user indicates the maximum time-step number for which the associated boundary condition card applies. In the example data set (Figure 15), the first line of boundary conditions would apply to the first 96 model time-steps. the second line for time-steps 97 through 120, etc. There must be NUM update intervals specified. The format for these cards is 7I10.

223. Boundary condition cards. There is one boundary conditions card for each update interval specified (a total of NUM, specified on the update card). The first piece of data is the time-step in seconds. Following this information are the values of the boundary conditions, given in the order indicated on the boundary conditions identification card. The maximum iteration number that a given boundary condition, or time-step, applies is indicated on the update interval card. The format is 7F10.0.

Output

224. Output from RIV1H consists of several files. One of the files, File 6, consists of the input data and the simulation results at specified print intervals. The simulation results contain the information for each segment at each time-step print interval, one segment per page. At the top of the page appears the program title. The next line prints the run title as supplied by the user. On the following lines are the time-step number and the elapsed time in the simulation, in hours and minutes followed by the segment ID number and name. Below that are column headings and data for river mile, flow (cubic feet per second), area (feet), width (feet), stage (feet), and water surface elevation (feet) (Figure 16).

225. Another output file is in standard FORTRAN unformatted records used as input to RIV1Q and is written to File 8. The records are organized as follows:

Record 1	MNODE,MTIME,NS
Record 2	JT,ITO,ITI,IT2,NNODE,NODE1,ID,JBCU
Record 3	DX1,QL,RMILE
Record 4	DT,Q,A,B,EL

where all variables are arrays except for MNODE, MTIME, NS, and DT. Record 4 is contained in the array HYDRO (with exceptions of DT) and is written every time-step, as this information is required to drive RIV1Q and plot hydrodynamic output. All succeeding records are identical to Record 4 but for succeeding time-steps. A diagnostics output file (File 7) is generated under certain conditions during the execution of RIV1H. If the Courant number becomes less than 1.0, if Manning's n becomes less than or equal to 0.01, or if the depth goes below 0.0, then a message is written to File 7 telling the user at which node this event occurred. If the program has trouble converging, a diagnostic message is written to File 7, allowing the user to examine the node at which nonconvergence is occurring.

Special considerations

226. Upland streams characterized by riffles and pools can be difficult to model with hydraulic routing models, such as RIV1H. Generally, two types of problems may be encountered: start-up of the model and phase errors

```
 SAMPLE DATA SET - USERS MANUAL
$CONST
BETA    =   1.000000    ,
GR      =   32.17000    ,
IPRINT  =         192,
RMILE0  =  0.0000000E+00,
THETA   =  0.6700000    ,
TOLER   =  0.1000000
$END
  4014.000   500.000   10.000   0.000   495.000   690.000   7.000   2.000   0.050   0.000   0.625   0.025
  5248.000   500.000   10.200   0.000   494.000   153.000   3.400   2.000   0.050   0.000   0.600   0.010
  3983.000   500.000   10.800   0.000   492.000    80.000   5.700   2.000   0.050   0.000   0.000   0.000
  4334.000   623.000   10.200   0.000   485.000   185.000   2.100   2.000   0.050   0.000   0.000   0.000
  4428.000   750.000   10.400   0.000   484.000   240.000   2.100   2.000   0.050   0.000   0.000   0.000
  4428.000   750.000   10.400   0.000   479.000   300.000   1.850   2.000   0.050   0.000   0.000   0.000
  4933.000   750.000   10.800   0.000   474.000   355.000   1.600   2.000   0.050   0.000   0.000   0.000
  3667.000   750.000   10.900   0.000   472.000   175.000   3.200   2.000   0.050   0.000   0.000   0.000
  4200.000   750.000   10.700   0.000   472.000   225.000   0.700   2.000   0.050   0.000   0.000   0.000
  4200.000   750.000   10.700   0.000   470.000   224.000   0.800   2.000   0.050   0.000   0.000   0.000
  4200.000   750.000   10.700   0.000   468.000   223.000   0.900   2.000   0.050   0.000   0.000   0.000
  4200.000   750.000   10.500   0.000   466.000   221.000   1.000   2.000   0.050   0.000   0.500   0.010
  4664.000   750.000   10.500   0.000   463.500   220.000   1.100   2.000   0.050   0.000   0.000   0.000
  4664.000   750.000   10.400   0.000   460.500   215.000   1.800   2.000   0.050   0.000   0.000   0.000
  4664.000   750.000   10.500   0.000   457.500   210.000   2.500   2.000   0.050   0.000   0.000   0.000
  4352.000   750.000   10.400   0.000   455.000   205.000   3.200   2.000   0.050   0.000   0.000   0.000
  4352.000   750.000   10.200   0.000   455.000   203.000   2.800   2.000   0.050   0.000   0.550   0.011
  4352.000   750.000   10.900   0.000   455.000   200.000   2.400   2.000   0.050   0.000   0.000   0.000
  4352.000   750.000   10.500   0.000   455.000   198.000   2.000   2.000   0.050   0.000   0.000   0.000
  3733.000   750.000   10.800   0.000   455.500   195.000   1.600   2.000   0.050   0.000   0.000   0.000
  3733.000   750.000   10.800   0.000   453.500   200.000   1.300   2.000   0.050   0.000   0.000   0.000
  3733.000   750.000   10.600   0.000   451.500   205.000   1.000   2.000   0.050   0.000   0.000   0.000
  4498.000   750.000   10.200   0.000   449.000   210.000   0.600   2.000   0.050   0.000   0.000   0.000
  4498.000   750.000   10.300   0.000   445.000   225.000   0.900   2.000   0.050   0.000   0.000   0.000
  4498.000   750.000   10.400   0.000   442.000   245.000   1.300   2.000   0.050   0.000   0.000   0.000
  4498.000   750.000   10.900   0.000   439.000   260.000   1.600   2.000   0.050   0.000   0.000   0.000
  5772.000   750.000   10.400   0.000   435.500   280.000   2.000   2.000   0.050   0.000   0.000   0.000
  5772.000   750.000   10.800   0.000   434.500   300.000   2.000   2.000   0.050   0.000   0.000   0.000
  5772.000   750.000   10.500   0.000   433.500   325.000   2.000   2.000   0.050   0.000   0.000   0.000
  5772.000   750.000   10.300   0.000   432.500   345.000   2.000   2.000   0.050   0.000   0.000   0.000
  5140.000   750.000   10.200   0.000   432.500   370.000   1.600   2.000   0.050   0.000   0.000   0.000
  5140.000   750.000   10.100   0.000   428.500   610.000   1.750   2.000   0.050   0.000   0.000   0.000
  5140.000   750.000   15.000   0.000   424.500   850.000   1.900   2.000   0.050   0.000   0.000   0.000
  4450.000   750.000   15.000   0.000   420.000  1100.000   2.100   2.000   0.050   0.000   0.000   0.000
  2564.000   250.000   10.500   0.000   488.500   177.000   3.000   2.000   0.040   0.000   0.500   0.010
  2970.000   250.000   10.200   0.000   488.000   140.000   1.900   2.000   0.040   0.000   0.000   0.000
  2970.000   250.000   10.500   0.000   484.800   143.000   2.100   2.000   0.040   0.000   0.000   0.000
```

Figure 16. Sample RIV1H output for a segment (Sheet 1 of 6)

```
2793.000   250.000    10.700     0.000    481.500   147.000   2.300   2.000   0.040   0.000   0.000   0.0
1977.000   250.000    10.700     0.000    482.500   212.000   1.800   2.000   0.050   0.000   0.000   0.0
 100.000   250.000    10.700     0.000    482.500   250.000   2.400   2.000   0.050   0.000   0.000   0.0
       1          0          0          0
      23
      96       120        144        164        184        200        220
     240       260        280        300        320        340        350
     370       390        400        420        440        460        480
     500       520
   450.      500.
   450.      500.
   450.      500.
   450.      600.
   450.      600.
   450.      600.
   450.      600.
   450.      700.
   450.      700.
   450.      800.
   450.      800.
   450.      700.
   450.      700.
   450.      700.
   450.      700.
   450.      600.
   450.      500.
   450.      400.
   450.      400.
   450.      400.
   450.      400.
   450.      400.
   450.      400.
```

Figure 16. (Sheet 2 of 6)

DYNAMIC RIVERINE WATER QUALITY MODEL,CE-QUAL-RIV1 RIV1H 7-SEP-89
SAMPLE DATA SET - USERS MANUAL
ELAPSED TIME 0 HOURS 0 MINUTES TIME STEP 0 SEGMENT 1 MAIN STEM ABOVE REREG DAM

	RIVER MILE	FLOW (CFS)	AREA (SQ FT)	WIDTH (FEET)	DEPTH (FEET)	WSEL (FEET)	MANNINGS N
1	27.76	500.000	7600.000	830.000	10.000	505.000	0.050
2	27.00	500.000	1914.336	222.360	10.200	504.200	0.050
3	26.01	500.000	1528.848	203.120	10.800	502.800	0.050
4	25.25	623.000	2105.484	227.840	10.200	495.200	0.050
5	24.43	750.000	2723.136	283.680	10.400	494.400	0.050
6	23.59	750.000	3320.096	338.480	10.400	489.400	0.050
7	22.76	750.000	4020.624	389.560	10.800	484.800	0.050
8	21.82	750.000	2287.692	244.760	10.900	482.900	0.050
9	21.13	750.000	2487.643	239.980	10.700	482.700	0.050
10	20.33	750.000	2488.392	241.120	10.700	480.700	0.050
11	19.54	750.000	2489.141	242.260	10.700	478.700	0.050
12	18.74	750.000	2430.750	242.000	10.500	476.500	0.050
13	17.94	750.000	2431.275	243.100	10.500	474.000	0.050
14	17.06	750.000	2430.688	252.440	10.400	470.900	0.050
15	16.18	750.000	2480.625	262.500	10.500	468.000	0.050
16	15.29	750.000	2478.112	271.560	10.400	465.400	0.050

DYNAMIC RIVERINE WATER QUALITY MODEL,CE-QUAL-RIV1 RIV1H 7-SEP-89
SAMPLE DATA SET - USERS MANUAL
ELAPSED TIME 0 HOURS 0 MINUTES TIME STEP 0 SEGMENT 2 MAIN STEM BELOW REREG DAM

	RIVER MILE	FLOW (CFS)	AREA (SQ FT)	WIDTH (FEET)	DEPTH (FEET)	WSEL (FEET)	MANNINGS N
1	15.29	750.000	2361.912	260.120	10.200	465.200	0.050
2	14.47	750.000	2465.144	252.320	10.900	465.900	0.050
3	13.65	750.000	2299.500	240.000	10.500	465.500	0.050
4	12.82	750.000	2292.624	229.560	10.800	466.300	0.050
5	12.11	750.000	2311.632	228.080	10.800	464.300	0.050
6	11.41	750.000	2285.360	226.200	10.600	462.100	0.050
7	10.70	750.000	2204.424	222.240	10.200	459.200	0.050
8	9.85	750.000	2412.981	243.540	10.300	455.300	0.050
9	9.00	750.000	2688.608	272.040	10.400	452.400	0.050
10	8.15	750.000	3024.096	294.880	10.900	449.900	0.050
11	7.29	750.000	3128.320	321.600	10.400	445.900	0.050
12	6.20	750.000	3473.280	343.200	10.800	445.300	0.050
13	5.11	750.000	3633.000	367.000	10.500	444.000	0.050
14	4.01	750.000	3765.680	386.200	10.300	442.800	0.050
15	2.92	750.000	3940.464	402.640	10.200	442.700	0.050
16	1.95	750.000	6339.518	645.350	10.100	438.600	0.050
17	0.97	750.000	13177.500	907.000	15.000	439.500	0.050
18	0.00	750.000	16972.500	1163.000	15.000	435.000	0.050

Figure 16. (Sheet 3 of 6)

DYNAMIC RIVERINE WATER QUALITY MODEL,CE-QUAL-RIV1 RIV1H 7-SEP-89
SAMPLE DATA SET - USERS MANUAL
ELAPSED TIME 0 HOURS 0 MINUTES TIME STEP 0 SEGMENT 3 TRIBUTARY UPSTRM OF REREG

	RIVER MILE	FLOW (CFS)	AREA (SQ FT)	WIDTH (FEET)	DEPTH (FEET)	WSEL (FEET)	MANNINGS N
1	2.51	250.000	2189.250	240.000	10.500	499.000	0.040
2	2.03	250.000	1625.676	178.760	10.200	498.200	0.040
3	1.47	250.000	1733.025	187.100	10.500	495.300	0.040
4	0.90	250.000	1836.227	196.220	10.700	492.200	0.040
5	0.37	250.000	2474.482	250.520	10.700	493.200	0.050
6	0.00	250.000	2949.776	301.360	10.700	493.200	0.050

DYNAMIC RIVERINE WATER QUALITY MODEL,CE-QUAL-RIV1 RIV1H 7-SEP-89
SAMPLE DATA SET - USERS MANUAL
ELAPSED TIME 24 HOURS 0 MINUTES TIME STEP 192 SEGMENT 1 MAIN STEM ABOVE REREG DAM

	RIVER MILE	FLOW (CFS)	AREA (SQ FT)	WIDTH (FEET)	DEPTH (FEET)	WSEL (FEET)	MANNINGS N
1	27.76	600.000	6469.651	810.697	8.622	503.622	0.409
2	27.00	578.517	1565.966	211.434	8.594	502.594	0.514
3	26.01	571.257	172.611	101.660	1.900	493.900	0.050
4	25.25	700.250	791.322	202.173	4.088	489.088	0.050
5	24.43	825.296	443.537	247.642	1.819	485.819	0.050
6	23.59	799.223	1515.356	318.141	4.903	483.903	0.050
7	22.76	751.643	3650.960	386.511	9.847	483.847	0.050
8	21.82	702.158	2518.456	250.721	11.831	483.831	0.050
9	21.13	673.681	2757.169	241.547	11.819	483.819	0.050
10	20.33	641.595	3246.357	246.098	13.811	483.811	0.050
11	19.54	609.070	3749.825	251.453	15.807	483.807	0.050
12	18.74	575.850	4240.214	256.519	17.759	483.759	0.322
13	17.94	541.530	4897.735	264.481	20.218	483.719	0.050
14	17.06	500.297	5962.081	298.583	23.218	483.718	0.050
15	16.18	453.552	7223.941	341.086	26.217	483.717	0.050
16	15.29	400.000	8525.888	388.788	28.717	483.717	0.050

DYNAMIC RIVERINE WATER QUALITY MODEL,CE-QUAL-RIV1 RIV1H 7-SEP-89
SAMPLE DATA SET - USERS MANUAL
ELAPSED TIME 24 HOURS 0 MINUTES TIME STEP 192 SEGMENT 2 MAIN STEM BELOW REREG DAM

	RIVER MILE	FLOW (CFS)	AREA (SQ FT)	WIDTH (FEET)	DEPTH (FEET)	WSEL (FEET)	MANNINGS N
1	15.29	400.000	1371.920	237.854	6.224	461.224	0.482
2	14.47	400.013	779.760	217.912	3.732	458.732	0.050
3	13.65	399.997	708.168	211.824	3.456	458.456	0.050
4	12.82	400.022	378.568	201.116	1.911	457.411	0.050
5	12.11	400.027	383.973	204.931	1.896	455.396	0.050
6	11.41	400.040	378.733	208.662	1.831	453.331	0.050

Figure 16. (Sheet 4 of 6)

7	10.70	400.049	328.955	211.871	1.560	450.560	0.050
8	9.85	400.057	347.451	227.763	1.535	446.535	0.050
9	9.00	400.092	449.463	249.724	1.817	443.817	0.050
10	8.15	400.126	323.511	263.952	1.235	440.235	0.050
11	7.29	400.472	604.293	288.504	2.126	437.626	0.050
12	6.20	402.010	636.155	308.365	2.091	436.591	0.050
13	5.11	404.731	755.942	334.174	2.294	435.794	0.050
14	4.01	407.688	1006.672	356.481	2.870	435.370	0.050
15	2.92	409.347	958.974	378.203	2.563	435.063	0.050
16	1.95	409.558	4040.204	632.757	6.502	435.002	0.050
17	0.97	409.501	9134.667	889.901	10.500	435.000	0.050
18	0.00	409.495	16972.500	1163.000	15.000	435.000	0.050

DYNAMIC RIVERINE WATER QUALITY MODEL,CE-QUAL-RIV1 RIV1H 7-SEP-89
SAMPLE DATA SET - USERS MANUAL
ELAPSED TIME 24 HOURS 0 MINUTES TIME STEP 192 SEGMENT 3 TRIBUTARY UPSTRM OF REREG

	RIVER MILE	FLOW (CFS)	AREA (SQ FT)	WIDTH (FEET)	DEPTH (FEET)	WSEL (FEET)	MANNINGS N
1	2.51	250.000	807.586	202.534	4.256	492.756	0.457
2	2.03	250.211	220.817	145.870	1.545	489.545	0.040
3	1.47	251.634	656.956	161.145	4.320	489.120	0.040
4	0.90	254.079	1250.033	181.962	7.600	489.100	0.040
5	0.37	256.331	1475.857	235.736	6.593	489.093	0.050
6	0.00	257.905	1751.091	281.627	6.588	489.088	0.050

DYNAMIC RIVERINE WATER QUALITY MODEL,CE-QUAL-RIV1 RIV1H 7-SEP-89
SAMPLE DATA SET - USERS MANUAL
ELAPSED TIME 48 HOURS 0 MINUTES TIME STEP 384 SEGMENT 1 MAIN STEM ABOVE REREG DAM

	RIVER MILE	FLOW (CFS)	AREA (SQ FT)	WIDTH (FEET)	DEPTH (FEET)	WSEL (FEET)	MANNINGS N
1	27.76	600.000	6867.046	817.557	9.110	504.110	0.397
2	27.00	664.041	1662.320	214.519	9.046	503.046	0.510
3	26.01	685.480	203.431	105.071	2.199	494.199	0.050
4	25.25	818.791	704.810	200.359	3.658	488.658	0.050
5	24.43	933.674	855.593	254.532	3.460	487.460	0.050
6	23.59	891.767	2566.782	330.142	8.147	487.147	0.050
7	22.76	833.399	4937.569	397.021	13.131	487.131	0.050
8	21.82	772.918	3378.609	271.792	15.124	487.124	0.050
9	21.13	737.952	3561.498	246.165	15.118	487.118	0.050
10	20.33	699.378	4067.751	251.382	17.114	487.114	0.050
11	19.54	659.916	4590.491	257.400	19.111	487.111	0.050
12	18.74	619.302	5104.770	263.173	21.087	487.087	0.289
13	17.94	577.262	5795.178	271.843	23.565	487.065	0.050
14	17.06	526.210	6981.607	310.633	26.565	487.065	0.050
15	16.18	467.489	8393.634	357.822	29.564	487.064	0.050
16	15.29	400.000	9863.106	410.211	32.064	487.064	0.050

Figure 16. (Sheet 5 of 6)

DYNAMIC RIVERINE WATER QUALITY MODEL,CE-QUAL-RIV1 RIV1H 7-SEP-89
 SAMPLE DATA SET - USERS MANUAL
ELAPSED TIME 48 HOURS 0 MINUTES TIME STEP 384 SEGMENT 2 MAIN STEM BELOW REREG DAM

	RIVER MILE	FLOW (CFS)	AREA (SQ FT)	WIDTH (FEET)	DEPTH (FEET)	WSEL (FEET)	MANNINGS N
1	15.29	400.000	1371.915	237.854	6.224	461.224	0.482
2	14.47	399.995	779.744	217.912	3.732	458.732	0.050
3	13.65	399.994	708.158	211.824	3.456	458.456	0.050
4	12.82	400.007	378.558	201.116	1.911	457.411	0.050
5	12.11	399.990	383.956	204.931	1.896	455.396	0.050
6	11.41	400.000	378.715	208.662	1.831	453.331	0.050
7	10.70	400.008	328.928	211.871	1.559	450.559	0.050
8	9.85	399.995	347.422	227.762	1.535	446.535	0.050
9	9.00	399.998	449.344	249.723	1.817	443.817	0.050
10	8.15	399.999	323.467	263.951	1.235	440.235	0.050
11	7.29	399.990	602.961	288.485	2.121	437.621	0.050
12	6 20	400.004	632.498	308.318	2.079	436.579	0.050
13	5.11	399.987	750.515	334.109	2.277	435.777	0.050
14	4.01	400.044	1002.164	356.430	2.857	435.357	0.050
15	2.92	399.959	957.989	378.195	2.561	435.061	0.050
16	1.95	399.988	4040.157	632.757	6.502	435.002	0.050
17	0.97	400.084	9134.678	889.901	10.500	435.000	0.050
18	0.00	400.000	16972.500	1163.000	15.000	435.000	0.050

DYNAMIC RIVERINE WATER QUALITY MODEL,CE-QUAL-RIV1 RIV1H 7-SEP-89
 SAMPLE DATA SET - USERS MANUAL
ELAPSED TIME 48 HOURS 0 MINUTES TIME STEP 384 SEGMENT 3 TRIBUTARY UPSTRM OF REREG

	RIVER MILE	FLOW (CFS)	AREA (SQ FT)	WIDTH (FEET)	DEPTH (FEET)	WSEL (FEET)	MANNINGS N
1	2.51	250.000	808.207	202.553	4.259	492.759	0.457
2	2.03	250.123	192.450	145.129	1.350	489.350	0.040
3	1.47	250.825	589.541	159.376	3.899	488.699	0.040
4	0.90	251.763	1172.761	179.994	7.173	488.673	0.040
5	0.37	252.761	1375.204	234.189	6.164	488.664	0.050
6	0.00	255.006	1630.475	279.553	6.158	488.658	0.050

Figure 16. (Sheet 6 of 6)

associated with unsteady flow waves. Streams with riffle sections accentuate these problems.

227. If inaccurate initial depths are selected for the initial flows, the model will have trouble getting started. If good initial estimates of depths are not available, it is best to run the model for a steady-state flow condition using deeper than expected conditions. With a constant inflow rate and either a constant head or rating curve downstream, the water surface in the segment will eventually drop to a steady-state water surface profile corresponding to that flow, channel geometry, and bed roughness. If problems are still encountered starting the model, the use of small time-steps, at least initially, has helped. After a steady-state flow and water surface profile has been achieved, it is best to use these results to start subsequent simulations.

228. Phase errors (see Constant Card, paragraphs 212-216), which may accompany unsteady flow waves, can result in negative depths (usually in or near shallow riffle sections). If this happens, a fatal execution error occurs, and the program is stopped. Several things can be done to eliminate this problem. First, make sure that the cross-sectional area is adequately described, especially at low stage. Next, check the value of THETA (see Constant Card, paragraphs 212-216); a higher value (e.g. 0.75) may help. If problems are still encountered, it may be necessary to change the time-step such that the surface wave Courant number is closer to 1.0. As values of this number decrease below or increase above 1.0, phase errors become worse (Lyn and Goodwin 1987, Liggett and Cunge 1975). The surface Courant number is defined as

$$C_N = \frac{U_s \Delta t}{\Delta x} \tag{209}$$

where

U_s = speed of the surface wave = $\sqrt{gd}$

d = water depth

Δt = time-step size

Δx = spatial step size

The same result can be achieved by adjustment of the spatial steps. If these measures are not successful, ramping of the inflow hydrograph to yield a gentler, longer wave will usually reduce phase errors since the unsteady flow waves are more spread out. As an example, hydropower turbines can come up to full power in about 5 min; peaking hydropower releases can result in highly unsteady flow waves in the receiving stream. It may be necessary to spread out the generation start-up over 15 to 30 min. Judicious smoothing of the inflow hydrograph can be accomplished without jeopardizing study objectives.

PART VIII: THE WATER QUALITY CODE (RIV1Q)

229. The water quality program contains sever subroutines in addition to the main program. The main program sets up the flexible dimensioning of arrays, as is done in the hydrodynamic program. Subroutine MAIN2 handles all input and output and the organization of the tributary structure. As noted in Part IV, RIV1Q achieves its fourth-order accuracy by advecting derivatives as well as concentrations. Initial concentrations are given by the input data, but initial derivatives must be estimated from these data. Subroutine MAIN2 calls subroutine SPLINE to pass a cubic spline through the data points, from which this estimate is obtained. The algorithm for constructing a cubic spline produces a tridiagonal matrix, which is solved by subroutine TRIDAG. At each step of the time march, MAIN2 proceeds through the tributary network in a downstream order (see Tributaries, paragraphs 189-192), assigning appropriate boundary conditions for each segment and calling subroutine SEG to complete the solution. At the completion of the last time-step, the program checks the input data stream for requests for sensitivity analyses. With each request, the entire simulation is rerun with new parameter values as indicated. At the exhaustion of sensitivity analyses, the program ends.

230. The major loops of the program are, in order of nesting, the sensitivity analysis, time, segment, node, and constituent. The program handles up to a maximum of 10 constituents, depending on the input: temperature (TEMP), CBOD, organic nitrogen (ORGAN), ammoniacal nitrogen (NH3-N), nitrate nitrogen (NO3-N), phosphate (PO4), DO, coliform bacteria, dissolved iron, and dissolved manganese (label, units, and kinetic constants supplied by the user). Minor loops of the program cycle through segments, nodes, or constituents as appropriate. A flowchart is shown in Figure 17.

Structure of RIV1Q

231. RIV1Q takes advantage of several non-ANSI FORTRAN features, such as namelist input and output and implicit type specification. All nonstandard features of this program are covered in Appendix A. Structured as a guided tour, this section is best read concurrently with the program listing in Appendix C.

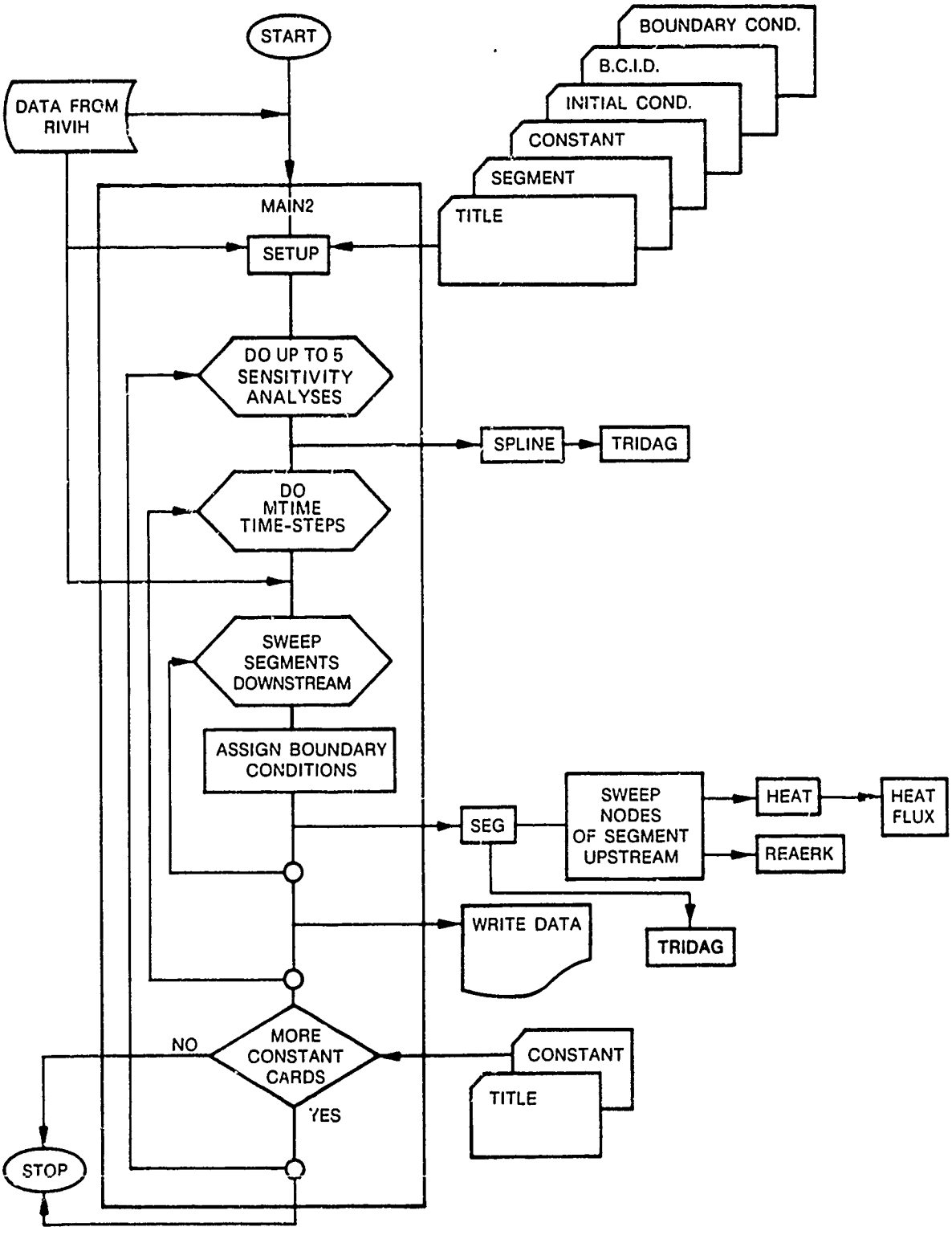


Figure 17. RIV1Q organizational flowchart

The MAIN program

232. The main program in RIV1Q serves to apportion storage in the same way as it does in RIV1H. The major difference is that whereas the grid information for the hydrodynamic program comes from the input deck, the water quality model obtains its grid information from the hydrodynamic model. Three variables determine the total array storage required by the program: IND1, the total number of computational nodes; IND2, the number of time-steps excluding initial conditions; and NS, the number of modeled segments. Total array storage held in the common storage array DJ is 103*NS + 67*IND1 + 10*NS*IND2 . The main program calls subroutine MAIN2 and terminates on return.

Subroutine MAIN2

233. Subroutine MAIN2 performs the bulk of the work for the rest of the program. It receives in its argument list the positions within the COMMON array DJ of all its flexibly dimensioned arrays. The structure of the argument list and dimension statement is such that several arrays are overlain in to optimize input. The input array HYDRO1 receives hydrodynamic information from RIV1H at odd time-steps and HYDRO2 at even time-steps. These input arrays are overlain by the individual hydrodynamic variables Q, A, B, and EL; for instance, the array Q is positioned such that Q(1) lies at the beginning of HYDRO1 and Q(IND1*4 + 1) lies at the beginning of HYDRO2. The multiplier of 4 in the index is to allow room for the arrays A, B, and EL. In this way, hydrodynamic data can be read in a block, but used individually.

234. Because of all the K-rates used, all variables whose names begin with the letter "K" are implicitly declared real. Label, units, and format arrays are declared double precision and initialized. A few parameters are explicitly declared real. The array CP, an array of flags to indicate the presence or absence of each of the 10 constituents, is declared logical and initialized to false. The DATA statements assign default values to parameters when reasonable, zero otherwise. The NAMELIST statement gives the list of variables whose value can be specified by namelist input.

235. The first item in the input stream is the title card. Then for each segment, the program reads the segment card, the constant card(s), and initial conditions cards. The computational nodes are stored in a 1-D array, even though they are handled segment by segment. The node number is initialized at 1, and the segment loop is entered. The segment card is read, and

then the constant card. On the constant card, the user indicates the name and value of any parameter whose default value is to be overridden. Most of these parameters can be stream-specific, but need not be. The program is set up so that the assignments on the first segment's constant card apply to every segment, but assignments on the constant cards of subsequent segments apply only to that segment. This setup is handled by looping through all the segments (1 to NS) for the first segment's card but only one (L to L) for subsequent segments. Each of these segment-specific parameters is packed into the array PARM. Lateral inflow concentrations go into CL. The parameter assignment section also takes care of conversion from customary American units (feet, seconds) to SI (metres, days). Based on lateral inflow concentrations or in situ sources, the presence/absence flags can be set to .TRUE. .

236. Following the constant card are the initial conditions cards, one for each node in the segment. The initial concentrations are stored in the array INIT so that they can be reused with each sensitivity analysis.

237. Not every segment receives boundary conditions from the input data stream; specifically, those segments below control structures receive their boundary conditions from the segment immediately above. Those segments that will receive their boundary conditions from the input data stream are indicated on the boundary conditions identification card in the order that the respective boundary conditions cards will appear. The program then reads in all the boundary conditions, storing them in the array BOUND for later reuse. It tests for the presence of each of the 10 constituents in the array BOUND.

238. The program refers to segments according to the order they appear in the input deck, but the user refers to each segment by its ID number. The logic must make the appropriate translation wherever such references occur. Similarly, it is convenient for the user to give the ID number of each of the segments for which boundary conditions are given; this information is stored in the array IBC. For the program, however, it is more convenient to know where the boundary conditions for each segment are to be found. This cross-reference type of information is given in the array JBC.

239. Distance increments (DX1), lateral inflows (QL), and river miles (RMILE) are time-invariant and so are read in from the hydrodynamic model outside of the time-march loop.

240. The date of program execution is called so that it can be printed on the output. The times of the start of the simulation, dawn, and sunset are

converted to HH.MM (hours and minutes) format to fractions of days; for instance, 7:12 p.m. would be given as 19.12 and converted to 0.8 days. Initial and boundary conditions are written out as they were read in, but without separation into segments.

241. The sensitivity-analysis loop is entered and is allowed a maximum of five passes. Once inside, the first task is to assign concentrations from the array INIT and test for the presence of each of the 10 constituents. This completes the presence tests that allow for the construction of the label, unit, and object-time format arrays. Labels are packed into the array BLABEL, and units, into the array BUNIT. The object-time format array is packed so that CBOD and DO are printed to one decimal place (corresponding to analytical accuracy); all other constituents to two places.

242. The running parameters of photoperiod, elapsed time, and clock time are initialized. If the simulation begins during daylight hours, the sine of the incident light is calculated for use in algal productivity calculations.

243. As mentioned previously, the fourth-order method requires initial derivatives as well as concentrations, although the user would have no a priori knowledge about them. Subroutine SPLINE is called to estimate derivatives at each point based on a cubic spline through the data. The spline is constructed one segment at a time.

244. Hydrodynamic data for the initial conditions are read and converted from customary American to SI units. The program calls subroutine SEG to set initial rate constants. Since actual transport and decay processes are not calculated at this point (this is indicated to the subroutine by setting DT = 0.), boundary conditions are not passed, and the ordering of the segments is unimportant.

245. The main time-march loop begins with setting MBC, the index to the boundary conditions array, for this time-step. The program determines if the time-step is odd or even and accordingly reads hydrodynamic data into HYDRO1 or HYDRO2 and sets the hydrodynamic array pointers L0 and L1. The solution algorithm in this fourth-order scheme is explicit, and so only one time level of information is required at any step. Hydrodynamic information, however, must be read in blocks, and two complete blocks must be available at every step.

246. To ensure proper passing of boundary conditions, subroutine SEG is called for each segment, working through the network in a downstream order. Boundary conditions are required for every segment, either from the input stream (JBC > 0) or passed down from another segment (JBC < 0). (JBC = 0 can happen only if boundary conditions are specified incorrectly. In this case, the program aborts.) In the first case, the program determines where in BOUND the boundary conditions for that segment are found and calls subroutine SEG with BOUND at that index. In the second case, the appropriate location in the tributary information transfer array, T, is given immediately by -JBC(L). The entry in the array T that stands for DO, i.e. T(7,L2), is adjusted for reaeration through control structures according to Equation 131. The program then calls subroutine SEG with boundary conditions given in T.

247. After processing all of the segments, the program prints out the data for all the segments, one to a page, in the order that the segments are given in the input deck. At the completion of the last time-step, the next title card is read, and the input deck is searched for new constant cards, as before, one per segment. If they are found, the parameters are assigned to the segments as described above, the hydrodynamic data file is rewound (and then advanced to pass over grid information, DX1, RMILE, and QL, which are all handled outside the sensitivity-analysis loop), and the simulation is rerun. If, on the other hand, the constant cards are not found, the program terminates.

Subroutine SEG

248. Subroutine SEG performs the main decay rate and transport and decay calculations for each segment. In its argument list are the distance increments, DX1; hydrodynamic data from the previous time-step, QO..., etc.; hydrodynamic data from the current time-step, Q..., etc.; lateral inflows, QL; concentrations, C; spatial derivatives of the concentrations, DC; decay rates, K; source/sink terms, SINK; boundary conditions, BC; presence/absence flags, CP; concentrations in the lateral inflow, CL; parameters, PARM; the sine of the incident light, SINI; the number of nodes, MNODE; the index of the first tributary to that segment within T, IT0; the index of the last tributary, IT1; the index where this segment is to place its data, IT2; the tributary information transfer arrays, JT and T; and the time increment, DT.

249. The first task is to assign data from the parameter array to scalars with more meaningful names. Next, a derivative at the first node is

estimated with a cubic interpolation based on the concentration at the boundary at the new time-step and at the first and second nodes at the old time-step, and the derivative at the second node. Markers to define the downstream extent of algal nutrient depletion are initialized.

250. In this explicit method, only two nodes from the old time-step are required in the solution of any node at the current time-step (Figure 18); that is, α_i^{j+1} is calculated from α_i^j and α_{i-1}^j. If calculation were to proceed in a downstream fashion, α_{i-1}^j would have to be saved so it would not be overwritten in the calculation of α_{i-1}^{j+1}. Proceeding upstream, however, removes this difficulty. Entries at i-1 are old information by default, and α_i^{j+1} simply replaces α_i^j as it is calculated. For this reason, the node loop marches backwards through the indices.

251. Once inside the node march loop, velocities and the rate constant adjustments for flow, QX (Q in Equation 111), are calculated. The

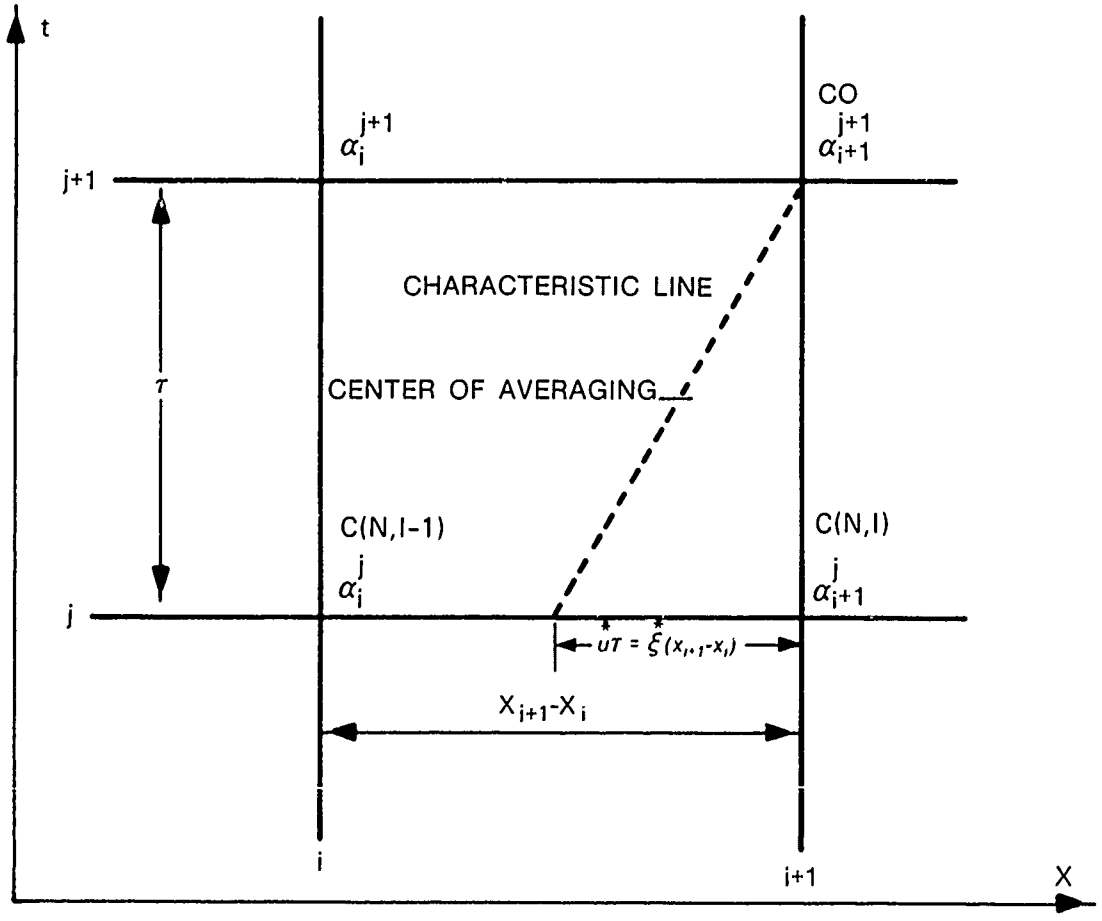


Figure 18. Compact fourth-order numerical gridwork

succeeding hydrodynamic manipulations are useful only in the transport calculations and so are bypassed for initial and boundary conditions. For interior grid points, these statements prepare the groundwork for the solution of the final transport Equations 92, 95, and 98. The distance increment DX is assigned to $X_{i+1} - X_i$ (the node to be calculated is termed node I in the program, as opposed to node i+1 in Part IV). For any parameter a, $\partial a/\partial x$ is calculated as $(a^j_{i+1} - a^j_i)/(X_{i+1} - X_i)$, $\partial a/\partial t$ is calculated as $(a^{j+1}_i - a^j_i)/\tau$, and the estimated average as

$$\overset{*}{a} = \frac{\left[a^{j+1}_{i+1} + a^j_{i+1}(1 - \xi) + a^j_i \xi\right]}{2} \tag{210}$$

The exception to this rule is u*, whose calculation must precede that of ξ and is given by Equation 92. The variable ξ is calculated by

$$\xi = \frac{u^* \tau}{X_{i+1} - X_i} \tag{211}$$

Its complement $(1 - \xi)$ is also assigned a variable. In the calculation of Equation 98, the following is the same for all constituents and so is assigned to the variable GO :

$$1 - \frac{\tau(u_{i+1} - u_i)}{X_{i+1} - X_i} \tag{212}$$

The coefficients A_1 through A_4 and b_1 through b_4 are calculated according to Equations 81 through 84 and 86 through 89, respectively, based on the estimated value of ξ and its complement.

252. With all the background to the transport equations prepared, attention shifts to the decay rates and source/sink terms. Decay rates, source/sink terms, and final concentrations and derivatives are calculated within a loop that cycles through each of the 10 modeled constituents. Decay rates and source/sink terms are particular to each constituent, and so each is calculated in a separate block. Once decay rates and source/sink terms for a particular constituent are calculated, the final averaging, concentration, and

derivative equations can be solved in a stereotyped way, and so the loop reassembles. (The reason that decay rates and source/sink terms are within the loop at all is the interdependence of constituents; for instance, the sink term for DO contains the final concentration of CBOD. If all decay rates and source/sink terms were calculated outside the loop, this information would not be available.) Finally the effect of diffusion is calculated implicitly and added to the constituent concentration.

253. The equations that appear in the listing are equivalent to those given in Parts IV and VI, with some minor modifications. In one instance, numerical problems presented by the expression

```
2./KEXT*(1-EXP(-KEXT*H))+B*EXP(-KEXT*H)                    (213)
```

used in determining available photosynthetic energy (Equation 156) must be handled. First, the number of calls to the external function EXP (exponentiation) can be cut in half by rewriting the above expression as the algebraically equivalent

```
2./KEXT + (B-2./KEXT)*EXP(-KEXT*H)                         (214)
```

The above expression, however, becomes numerically unstable as KEXT approaches zero, although mathematically it approaches the value

```
2.*H + B                                                   (215)
```

This problem is handled by taking a Taylor's series expansion of EXP(-KEXT*H) in the expression 2./KEXT*(1.-EXP(-KEXT*H)) about KEXT = 0 carried to three terms,

```
2./KEXT*(1. - (1. - KEXT*H + ((KEXT*H)**2)/2 + ....))
```

This simplifies to

```
H*(2. - KEXT*H)                                            (216)
```

The entire expression is then

$$H*(2. - KEXT*H) + B*EXP(-KEXT*H) \tag{217}$$

for KEXT less than 0.01.

254. After calculating decay rates and source/sink terms, the loop reassembles to calculate averages and spatial derivatives of these quantities. Finally, the concentrations and their spatial derivatives are calculated.

255. Lateral inflow is modeled by simple mass balance. Subtracting the continuity equation (Equation 34) times the concentration, α , from the pollutant transport equation (Equation 36), yields (neglecting off-channel storage, diffusion, decay, and source/sink terms)

$$\frac{\partial \alpha}{\partial t} + U \frac{\partial \alpha}{\partial x} = \frac{q}{A} (\gamma - \alpha) \tag{218}$$

where q is the lateral inflow (L^2T^{-1}) and γ is the concentration in the lateral inflow. The quantity q/A is denoted as QLA and $\gamma - \alpha$ is replaced by CCL. The right-hand side of the transport equation is therefore augmented by the term

$$CCL*QLA \tag{219}$$

The derivative transport equation receives this term after product rule differentiation

$$(-DC*QLA + CCL*DQLA) \tag{220}$$

where DC and DQLA are, respectively, αx and $\partial(q/A)/\partial x$.

256. Once the calculations for all the constituents are complete, the frame of interest shifts by one node, and iteration continues.

257. At the completion of the node march, the boundary conditions are assigned to the first node, the segment loads its own information into T, and the algal nutrient depletion data are placed in PARM(25) through PARM(28).

Subroutine SPLINE

258. As stated earlier, this solution scheme requires spatial derivatives for the concentrations of all modeled constituents at initial and boundary conditions. Because the user would have no a priori information as to the values of the derivatives, the program estimates them based on a polynomial interpolation of the initial and boundary data themselves. The scheme's being fourth order suggests that interpolation should likewise be fourth order, that is, a cubic spline. Such a spline is completely defined by the following considerations: the curve should pass through every data point; it should be fourth order; although the equation may vary from one interval to the next, the curve and its first and second derivatives should be continuous at every point; the second derivative at the two boundaries should be zero (that is, there should be no bending of the curve at the end points, commonly known as a "relaxed" or "natural" condition).

259. A complete development of the equations is given by Rogers and Adams (1976). Briefly, the cubic equation for each interval is defined by the values and derivatives at each end of the interval, just as is done in the development of the fourth-order scheme in Part IV. Requiring that the second derivative be continuous at every point implies that the cubic equations for any two adjacent intervals yield equal second derivatives at their point of juncture. This leads to a system of $n - 2$ linear equations, where n is the number of nodes, that is, one equation for each interior node. The system is completed by the equation that sets the second derivative to zero at the first node and the one that does the same for the last node. This produces a system of n linear equations in n unknowns, where the coefficient matrix is of tridiagonal form filled with nonzero members along the main diagonal, one row above the main diagonal and one below. Tridiagonal matrices are amenable to much faster solution than general square matrices--the solution scheme employed here is subroutine TRIDAG, taken with slight modification, from Carnahan, Luther, and Wilkes (1969).

260. The mathematical formalism proceeds as follows. Recall Equation 80 in expanded form:

$$Y(\xi) = \xi^2(3 - 2\xi)\alpha_i + [1 - \xi^2(3 - 2\xi)]\alpha_{i+1}$$

$$+ \xi^2(1 - \xi)(x_{i+1} - x_i)\ \alpha x_i - \xi(1 - \xi)^2(x_{i+1} - x_i)\alpha x_{i+1} \qquad (221)$$

The expression for the second derivative is then:

$$\ddot{Y}(\xi) = (6 - 12\xi)\alpha_i + (-6 + 12\xi)\alpha_{i+1}$$

$$+ (2 - 6\xi)(x_{i+1} - x_i)\alpha x_i + (4 - 6\xi)(x_{i+1} - x_i)\alpha x_{i+1} \qquad (222)$$

Approaching node i+1 from the left, $\xi = 0$, and approaching it from the right $\xi = 1$. The requirement that the second derivative be continuous translates mathematically to

$$\left.\frac{d^2y}{dx^2}\right|_{x_{i+1}^-} = \left.\frac{d^2y}{dx^2}\right|_{x_{i+1}^+} \qquad (223)$$

Now

$$\left.\frac{d^2y}{dx^2}\right|_{x_{i+1}^-} = \left.\ddot{Y}(\xi)\right|_{\xi=0} \cdot \frac{1}{\left(x_i - x_{i+1}\right)^2} \qquad (224)$$

and

$$\left.\frac{d^2y}{dx^2}\right|_{x_{i+1}^+} = \left.\ddot{Y}(\xi)\right|_{\xi=1} \cdot \frac{1}{\left(x_{i+1} - x_{i+2}\right)^2} \qquad (225)$$

Equating the two:

$$\frac{1}{\left(x_i - x_{i+1}\right)^2}\ddot{Y}(\xi)\Big|_{\xi=0} = \frac{1}{\left(x_{i+1} - x_{i+2}\right)^2}\ddot{Y}(\xi)\Big|_{\xi=1} \qquad (226)$$

which can also be written:

$$\frac{x_{i+2} - x_{i+1}}{x_{i+1} - x_i}\ddot{Y}(\xi)\Big|_{\xi=0} = \frac{x_{i+1} - x_i}{x_{i+2} - x_{i+1}}\ddot{Y}(\xi)\Big|_{\xi=1} \qquad (227)$$

In expanded form:

$$\frac{x_{i+2} - x_{i+1}}{x_{i+1} - x_i}\left[6\alpha_i - 6\alpha_{i+1} + 2(x_{i+1} - x_i)\alpha x_i + 4(x_{i+1} - x_i)\alpha x_{i+1}\right]$$

$$= \frac{x_{i+1} - x_i}{x_{i+2} - x_{i+1}}\left[-6\alpha_{i+1} + 6\alpha_{i+2} - 4(x_{i+2} - x_{i+1})\alpha x_{i+1} - 2(x_{i+2} - x_{i+1})\alpha x_{i+2}\right] \tag{228}$$

Rearranging so that all the derivatives are on the left-hand side and everything else on the right-hand side and dividing by two yields:

$$(x_{i+2} - x_{i+1})\alpha x_i + 2\left[(x_{i+2} - x_{i+1}) + (x_{i+1} - x_i)\right]\alpha x_{i+1} + (x_{i+1} - x_i)\alpha x_{i+2}$$

$$= 3\left[\frac{x_{i+2} - x_{i+1}}{x_{i+1} - x_i}(\alpha_{i+1} - \alpha_i) + \frac{x_{i+1} - x_i}{x_{i+2} - x_{i+1}}(\alpha_{i+2} - \alpha_{i+1})\right] \tag{229}$$

The second derivative being zero at the first node requires (setting $\xi = 1$) that

$$-6\alpha_1 + 6\alpha_2 - 4(x_2 - x_1)\alpha x_1 - 2(x_2 - x_1)\alpha x_2 = 0 \tag{230}$$

or

$$\alpha x_1 + \frac{1}{2}\alpha x_2 = \frac{3}{2}\frac{1}{(x_2 - x_1)}(\alpha_2 - \alpha_1) \tag{231}$$

Similarly for the last node

$$\frac{1}{2}\alpha x_{n-1} + \alpha x_n = \frac{3}{2}\frac{1}{(x_n - x_{n-1})}(\alpha_n - \alpha_{n-1}) \tag{232}$$

261. The entire system can be written in matrix form and given to subroutine TRIDAG to solve.

$$\begin{bmatrix} 1 & \frac{1}{2} & & \\ & (x_{i+2} - x_{i+1}) & 2[(x_{i+2} - x_{i+1}) + (x_{i+1} - x_i)](x_{i+1} - x_i) & \\ & & \frac{1}{2} & 1 \end{bmatrix} \cdot \begin{bmatrix} \alpha x_i \\ \alpha x_{i+1} \\ \alpha x_n \end{bmatrix}$$

$$= \begin{bmatrix} \frac{3}{2} \frac{1}{(x_2 - x_1)} (\alpha_2 - \alpha_1) \\ 3\left[\frac{x_{i+2} - x_{i+1}}{x_{i+1} - x_i} (\alpha_{i+1} - \alpha_i) + \frac{x_{i+1} - x_i}{x_{i+2} - x_{i+1}} (\alpha_{i+2} - \alpha_{i+1})\right] \\ \frac{3}{2} \frac{1}{(x_n - x_{n-1})} (\alpha_n - \alpha_{n-1}) \end{bmatrix} \quad (233)$$

Tributaries

262. Tributaries in the constituent transport equations are considered to be point additions of materials. The concentration at the junction point is a function of the concentration and flow of the tributary and the receiving stream just upstream of the junction. For a junction at node i :

$$\alpha_i = \frac{(\alpha\emptyset_i \cdot Q\emptyset_i + \alpha_T \cdot Q_T)}{(Q\emptyset_i + Q_T)} \quad (234)$$

where

α = concentration

$\emptyset$ = receiving stream just upstream of the junction

Q = flow

T = tributaries

RIV1Q requires two reaches to achieve full dilution of the tributary flow; therefore Q_{i+1} replaces the sum of $Q\emptyset$ and Q_T in Equation 234,

$$\alpha_i = \frac{[\alpha\emptyset_i (Q_{i+1} - Q_T) + \alpha_T \cdot Q_T]}{Q_{i+1}}$$

$$= \alpha\emptyset_i \left(1 - \frac{Q_T}{Q_{i+1}}\right) + \alpha_T \left(\frac{Q_T}{Q_{i+1}}\right) \quad (235)$$

The quantity Q_T/Q_{i+1} can be represented by the dilution ratio D :

$$\alpha_i = \alpha\phi_i \ (1 - D) + \alpha_T \ D \tag{236}$$

263. To preserve fourth-order correctness through a junction, derivatives must be advected as well. The derivatives are diluted in the same way as concentrations, with the addition that spatial derivatives on the tributary must be adjusted for the difference between the velocity on the tributary and on the receiving stream:

$$\alpha x_i = \frac{\left[\alpha x\phi_i \ (Q_{i+1} - Q_T) + \alpha x_T \ Q_T \ \frac{u_T}{u_i}\right]}{Q_{i+1}}$$

$$= \alpha x\phi_i \left(1 - \frac{Q_T}{Q_{i+1}}\right) + \left(\alpha x_T \ \frac{u_T}{u_i}\right) \ \frac{Q_T}{Q_{i+1}}$$

$$= \alpha x\phi_i \ (1 - D) + \alpha x_T \ DD \tag{237}$$

where DD represents the quantity $D \cdot u_T/u_i$.

264. Tributary influx is modeled here as a discontinuity, and so the values immediately upstream of the junction point must be stored. The first 20 members of the T-array contain concentrations and their derivatives at the mouth. Location 21 contains tributary flow, and location 22 contains tributary velocity. Locations 23 through 32 contain concentrations on the main stem just upstream of the junction point, and locations 33 through 42 contain the derivatives there. In the solution procedure for the receiving stream, each node is checked to see if it is a junction point. If it is not, the solution proceeds normally. If it is, the array pointers shift to the appropriate positions in the T-array and $\alpha\phi$ and $\alpha x\phi$ are used in place of α_i and αx_i in performing advection and first-order decay. The calculated values become $\alpha\phi$ and $\alpha x\phi$ for the new time-step. Finally, α_i and αx_i are calculated according to Equations 236 and 237.

Input and Output Formats

Input

265. Input for RIV1Q consists of card (or card-image) input from the user and unformatted disk (or equivalent) input from RIV1H (or similar hydraulics model). Card input is composed of eight types of cards: title, segment, constant, initial conditions, boundary conditions identification, update, update intervals, and boundary conditions. The title card supplies the title with which the output of each run is labeled. The segment card gives data pertinent to each segment, such as name and identification number. On the constant card, rate coefficients and the other parameters are given values by name. The initial conditions cards and boundary conditions cards assign concentrations to each of the modeled constituents at every node for the first time-step, in the case of initial conditions, and at every appropriate boundary for each update time in the case of boundary conditions. The segments for which boundary conditions are given are listed on the boundary conditions identification card. If the user does not want to simulate a particular constituent (such as reduced iron), then the initial and boundary conditions and lateral loadings should be set to zero or left blank for that constituent.

266. The organization of the input deck is shown in Figure 19. The title card comes first, followed by a segment card, constant cards, and initial conditions cards for each segment. Boundary conditions are introduced by a boundary conditions identification card and followed by the update card, update interval cards, and the boundary conditions cards. This is the complete deck for making a single run. To perform sensitivity analyses, however, this deck is followed by a new title card and a batch of constant cards, one set of constant cards per segment. The process is repeated for each sensitivity analysis requested, up to a limit of five. The program ends upon exhaustion of the input.

267. Title card. The user has the full 80 columns of the title card to write any title of his choosing. It will be printed, verbatim near the top of every page.

268. Segment card. The format of the segment card is I2,10A4,I2,F10.0. The items specified are ID, the identification number for this segment, SNAME, its name, IDAM, and DAMK. The two latter variables concern reaeration that

```
 SAMPLE DATA SET - USERS MANUAL
 1 MAIN STEM ABOVE REREG DAM                    1     0.045
 &CONST START = 16.00, AK1 = 0.2, QWINDO = TRUE, IPRINT = 192,
 TEMPL = 25.0, ITEM = 0, &END
23.4      0.01     0.01      0.1    0.01     0.02     8.50    0.0      0.0       0.0
250.
22.8      0.01     0.01      0.1    0.01     0.02     8.60    0.0      0.0       0.0
250.
22.8      0.01     0.01      0.1    0.01     0.02     8.70    0.0      0.0       0.0
250.
24.3      0.01     0.01      0.1    0.01     0.02     8.20    0.0      0.0       0.0
250.1
24.4      0.01     0.01      0.1    0.01     0.02     8.20    0.0      0.0       0.0
250.
24.5      0.01     0.01      0.1    0.01     0.02     8.30    0.0      0.0       0.0
250.
24.7      0.01     0.01      0.1    0.01     0.02     8.30    0.0      0.0       0.0
250.
25.1      0.01     0.01      0.1    0.01     0.02     8.30    0.0      0.0       0.0
250.
25.4      0.01     0.01      0.1    0.01     0.02     8.30    0.0      0.0       0.0
250.
25.7      0.01     0.01      0.1    0.01     0.02     8.20    0.0      0.0       0.0
250.
25.9      0.01     0.01      0.1    0.01     0.02     8.20    0.0      0.0       0.0
250.
26.0      0.01     0.01      0.1    0.01     0.02     8.20    0.0      0.0       0.0
250.
26.0      0.01     0.01      0.1    0.01     0.02     8.20    0.0      0.0       0.0
250.
25.9      0.01     0.01      0.1    0.01     0.02     8.20    0.0      0.0       0.0
250.
25.7      0.01     0.01      0.1    0.01     0.02     8.30    0.0      0.0       0.0
250.
25.3      0.01     0.01      0.1    0.01     0.02     8.40    0.0      0.0       0.0
250.
 2 MAIN STEM BELOW REREG DAM
 &CONST &END
24.8      0.01     0.01      0.1    0.01     0.02     8.50    0.0      0.0       0.0
250.
24.3      0.01     0.01      0.1    0.01     0.02     8.50    0.0      0.0       0.0
250.
23.9      0.01     0.01      0.1    0.01     0.02     8.60    0.0      0.0       0.0
250.
23.8      0.01     0.01      0.1    0.01     0.02     8.60    0.0      0.0       0.0
250.
23.7      0.01     0.01      0.1    0.01     0.02     8.60    0.0      0.0       0.0
250.
23.8      0.01     0.01      0.1    0.01     0.02     8.50    0.0      0.0       0.0
250.
24.0      0.01     0.01      0.1    0.01     0.02     8.50    0.0      0.0       0.0
250.
25.0      0.01     0.01      0.1    0.01     0.02     8.40    0.0      0.0       0.0
250.
25.6      0.01     0.01      0.1    0.01     0.02     8.30    0.0      0.0       0.0
```

Figure 19. Sample RIV1Q input (Continued)

```
250.
26.0      0.01      0.01      0.1      0.01      0.02      8.30      0.0      0.0      0.0
250.
25.4      0.01      0.01      0.1      0.01      0.02      8.30      0.0      0.0      0.0
250.
24.2      0.01      0.01      0.1      0.01      0.02      8.50      0.0      0.0      0.0
250.
23.4      0.01      0.01      0.1      0.01      0.02      8.50      0.0      0.0      0.0
250.
23.7      0.01      0.01      0.1      0.01      0.02      8.40      0.0      0.0      0.0
250.
23.9      0.01      0.01      0.1      0.01      0.02      8.20      0.0      0.0      0.0
250.
23.9      0.01      0.01      0.1      0.01      0.02      8.10      0.0      0.0      0.0
250.
23.9      0.01      0.01      0.1      0.01      0.02      8.10      0.0      0.0      0.0
250.
23.9      0.01      0.01      0.1      0.01      0.02      8.10      0.0      0.0      0.0
250.
 3 TRIBUTARY UPSTRM OF REREG
 &CONST &END
25.0      0.01      0.01      0.1      0.01      0.02      7.20      0.0      0.0      0.0
250.
25.1      0.01      0.01      0.1      0.01      0.02      7.40      0.0      0.0      0.0
250.
25.4      0.01      0.01      0.1      0.01      0.02      7.50      0.0      0.0      0.0
250.
26.2      0.01      0.01      0.1      0.01      0.02      7.50      0.0      0.0      0.0
250.
26.3      0.01      0.01      0.1      0.01      0.02      7.60      0.0      0.0      0.0
250.
26.2      0.01      0.01      0.1      0.01      0.02      7.60      0.0      0.0      0.0
250.
+1,+3/
       4
     155       215       400       520
    12.0       1.0       0.5       0.2       0.1       0.05      2.5
    15.0       1.0       0.5       0.2       0.1       0.05      4.0
    12.0       1.0       0.5       0.2       0.1       0.05      2.5
    15.0       1.0       0.5       0.2       0.1       0.05      4.0
       1
     520
    20.0       1.0       0.1       0.01      0.1       0.01      8.0
```

Figure 19. (Concluded)

occurs through the control structure, if any, at the top of this segment. If this segment is not headed by a control structure or if reaeration is not modeled, these two fields are left blank. If IDAM is 0 or blank, structural reaeration is not modeled. If IDAM is given a value of one or greater, structural reaeration is modeled with the energy dissipation model (Wilhelms and Smith 1981),

$$r = e^{-C_T \Delta H} \tag{238}$$

where

r = deficit ratio (final DO deficit/initial DO deficit)

C_T = escape coefficient

ΔH = difference in water surface elevation across the control structure

The value for the escape coefficient at 20° C is input by DAMK with units of ft^{-1}; the program corrects for temperature by,

$$C_T = \text{DAMK}*1.022^{(\text{TEMP}-20)} \tag{239}$$

A value of 0.045 ft^{-1} for DAMK was recommended (Wilhelms and Smith 1981) for gated-conduit outlet works and low head weirs and gated spillways (H < 25 ft) with free hydraulic jumps. The value for this coefficient may be significantly different for other types of outlet structures or for conditions not conducive to reaeration, such as submerged hydraulic jumps that can be experienced with many low-sill gated structures. The program is not set up to allow structural reaeration at headwater nodes.

269. Constant card. On this card, the user specifies values by name for parameters and coefficients used in the model. This card also lies at the heart of the sensitivity-analysis feature. A constant card must be given for every segment, even if no values are specified on it. The program will scan through the deck looking for it and terminate if it fails to find it. There are 43 variables that can receive new values on the constant card. Their names and uses are given in Table 4. Any parameters that apply to the network as a whole (e.g., time of sunrise or a rate coefficient whose value is the same everywhere) are given in the constant card of the first segment.

Table 4

Parameters That Can Appear on the Constant Card

Name	Units	Default	Explanation
ADN	day^{-1}	0.1	Rate coefficient for denitrification.
AG	$day^{-1} fps^{-E1}/ft^{-E2}$	12.81	Rate coefficient for stream reaeration rate in the form $K_2 = \frac{AG\ U^{E1}}{H^{E2}}$. The default value is for the O'Connor-Dobbins (1958) formulation.
AKN	day^{-1}	0.3	Rate coefficient for nitrification.
AKNX	day^{-1}	0.0	Rate coefficient for ammonia adsorption.
AK1	day^{-1}	0.15	Rate coefficient for CBOD decay and organic nitrogen decay.
ALGAEB	g/m^2	0.0	The amount of benthic algae.
ATB	day^{-1}	0.0	Rate coefficient for bottom heat exchange.
ATS	$\frac{W^*}{m^2\ °C}$	0.0	Rate coefficient for surface heat exchange in constant equilibrium temperature method.
AP04	day^{-1}	0.0	Rate coefficient for phosphate adsorption.
CBODL	mg/ℓ	0.0	Concentration of CBOD in the segment's lateral inflows.

(Continued)

* $1 \frac{W}{m^2\ °C} = 4.23 \frac{BTU}{ft^2\ day\ °F}$

Table 4 (Continued)

Name	Units	Default	Explanation
COLIL	mg/ℓ	0.0	Coliform bacteria concentration in the segment's lateral inflows.
DAWN	HH.MM	6.30	Time of sunrise.
DOL	mg/ℓ	0.0	DO concentration in the segment's lateral inflows.
E1	Unitless	0.50	Exponent of velocity in stream reaeration rate formulation in the form $K_2 = \frac{AG\ U^{E1}}{H^{E2}}$. Default value is for the O'Connor-Dobbins (1958) formulation.
E2	Unitless	1.50	Exponent of depth in stream reaeration rate formulation in the form $K2 = \frac{AG\ U^{E1}}{H^{E2}}$. Default value is for the O'Connor-Dobbins (1958) formulation.
FEL	mg/ℓ	0.0	Reduced iron concentration in the segment's lateral inflows.
HNEFSW	W/m^2	0.0	Surface light intensity at local noon. A value is required if modeling photosynthesis but not using the direct energy budget that computes HNEFSW.
IPRINT		0	The output print interval (e.g. IPRINT = 10 causes only every tenth time-step to be printed to File 6).

(Continued)

Table 4 (Continued)

Name	Units	Default	Explanation
IPLOT		0	The output graphics print interval (e.g. IPLOT = 10 causes only every tenth time-step to be printed to the graphics File 19). For IPLOT = 0, no graphics output file is created.
ITEM		0	To designate type of heat exchange solution. For ITEM = 1 , the constant equilibrium temperature solution is used (should also specify ATS and TEMP). For ITEM = 0 , the direct energy balance is used for temperature solution, and meteorological data must be furnished. For any other value of ITEM other than 0 or 1, heat exchange is not modeled.
KALGDK	day^{-1}	0.1	Algal decay rate.
KALGRO	m^2 $Watts^{-1}$ day^{-1}	0.0020	Algal growth rate.
KCOLIDK	day^{-1}	1.4	Rate coefficient for coliform bacteria mortality.
KEXT	ft^{-1}	0.5	Light extinction coefficient.
KFEDK	day^{-1}	0.5	Rate coefficient for iron oxidation.
KMNDK	day^{-1}	0.5	Rate coefficient for manganese oxidation.
KNCBDN	mg/ℓ	0.1	Nitrate half-saturation constant for denitrification.
KOALDK	mg/ℓ	0.5	DO half-saturation constant for algal decay.

(Continued)

Table 4 (Continued)

Name	Units	Default	Explanation
KOCBDN	mg/ℓ	0.5	DO half-saturation constant for denitrification.
KOCB1	mg/ℓ	0.5	DO half-saturation constant for CBOD decay (Hoover and Porges 1952).
KON	mg/ℓ	0.5	DO half-saturation constant for nitrification.
MNL	mg/ℓ	0.0	Reduced manganese concentration in the segment's lateral inflows.
NH3NL	mg/ℓ	0.0	Ammonia concentration in the segment's lateral inflows.
NO3NL	mg/ℓ	0.0	Nitrate concentration in the segment's lateral inflows.
ORGANL	mg/ℓ	0.0	Organic nitrogen concentration in the segment's lateral inflows.
OXIDAT	mg/ℓ	1.0	Concentration of DO at and below which iron and manganese oxidation do not occur.
PO4L	mg/ℓ	0.0	Phosphate concentration in the segment's lateral inflows.
QWINDO		FALSE	A logical variable to invoke wind-driven reaeration. To turn on wind reaeration for a segment, set QWINDO = TRUE .
START	HH.MM	00.00	Starting time of the simulation.
SUNSET	HH.MM	20.30	Time of sunset.

(Continued)

Table 4 (Concluded)

Name	Units	Default	Explanation
TEMP	°C	23.0	If temperature is not modeled but temperature dependent processes are modeled, then a stream segment temperature should be specified with TEMP. For this condition, temperature values for initial conditions, headwater boundary conditions, and lateral inflows should not be specified; otherwise temperatures will be modeled by the heat exchange method selected by ITEM. If temperature is modeled with the constant equilibrium temperature approach, then TEMP is used to input the equilibrium temperature for each stream segment.
TEMPL	°C	0.0	Temperature of the segment's lateral inflows.
TSINK	°C	10.0	Source/sink term for bottom heat exchange.
TSIV	ft^{-1}	0.0	Coefficient in the Tsivoglou-Wallace reaeration formula (a value of zero serves to indicate that the equation $K2 = \frac{AG\ U^{E1}}{H^{E2}}$ will be used rather than this). If the Tsivoglou-Wallace formula is used, a value of 0.054 is suggested for TSIV.

Segment-specific exceptions to these values are given on the constant cards of the individual segments.

270. The form of the constant card is as follows. The first space is left blank. Columns 2-7 contain the characters &CONST. Following that are names of the variables, each followed by an equals sign followed by the value the user wishes to assign. If the user is content with the default value, then that variable need not appear on the constant card. Assignments are separated by commas and may extend over as many cards as necessary without regard to card field (the first column of every succeeding card must be left blank). At the end of the list goes the expression &END. (If the user misspells the name of a variable, in most systems the computer will print an error message and continue execution with the default value.)

271. Initial conditions cards. Two cards are provided for each node. The first card contains the initial conditions, and the second contains the horizontal dispersion coefficient. As stated in Part IV, necessary and sufficient initial conditions are the concentrations of all the parameters to be modeled. Accordingly, each initial conditions card is divided into 10 areas, each eight columns wide, which contain the values of temperature, CBOD, organic nitrogen, ammonia nitrogen, nitrate plus nitrate nitrogen, phosphate, dissolved oxygen, reduced manganese, reduced iron, and coliform bacteria, respectively. Columns corresponding to constituents that are not modeled are left blank or set to zero. Additionally, lateral inflow values (constant card) should not be specified for constituents that are not to be modeled. Each initial conditions card is followed by a card specifying the longitudinal dispersion, specified in the first eight columns. The format for these cards is 10F8.0.

272. The user can simulate an arbitrary conservative or nonconservative constituent with the coliform bacteria variable by setting KCOLIDK = 0.0 or to a specified first-order kinetic rate, respectively.

273. In most applications, the initial concentrations of water quality constituents at each node will not be known. This problem can be overcome by several means. If water quality data are available at several stations along the stream, values could be interpolated for the nodes. If such data are not available for the conditions being simulated, then a simulation can be made with estimated initial conditions and time-invariant boundary conditions that are the same as the first set or time-variant boundary conditions. The

resulting steady-state nodal concentrations are then used as the initial conditions for subsequent simulations. However, these steady-state initial conditions still may not be representative. If the nodal concentrations change significantly following the first flush (system travel time), then the initial conditions were probably not representative, and the results during the first flush should not be used for interpretation.

274. It is possible to improve the initial condition estimates to provide more meaningful results during first flush if conditions (flows, loadings, etc.) during or at the end of the simulation (but after first flush) are similar to the starting conditions. A dynamic simulation is made with estimated initial conditions; the simulated nodal concentrations, at the time that conditions (flows, loadings, etc.) are similar to initial conditions, are selected for initial conditions in subsequent simulations. For example, suppose that the simulation starts on Monday and ends on Sunday for a stream below a peaking hydropower dam where peaking operations occur on weekdays and steady low flows are maintained on weekends. Then the simulated nodal concentrations on Sunday should provide reasonable estimates for the initial Monday conditions (provided sufficient time elapsed for first flush and other factors are similar at the beginning and end of the simulation).

275. Boundary conditions identification card. The transport equations require boundary conditions for the upstream end of every segment. However, in the case where the segments are actually part of the same stream but are separated by a control structure, two segments are modeled but the boundary conditions for the lower segment are determined by the upper segment and do not appear in the input. The type of boundary conditions required, therefore, is uniquely determined once the network structure is fixed. The user indicates to the program on this card the segments (ID) whose boundary conditions will be given. The numbers can appear anywhere on the card and can be separated by a comma or blanks or both, but they must appear in order, with the list terminated by a slash (/).

276. Update card. The user here indicates the total number (NUM) of different boundary condition updates for the first segment identified on the boundary conditions identification card. The format for this card is I8. This card is then followed by the update interval cards and the boundary condition cards described below. A total of NUM boundary condition update intervals and NUM boundary conditions cards must be provided. The sequence is

then repeated for each of the segments specified on the boundary conditions identification card.

277. Update interval cards. The user here indicates the maximum iteration number for which the associated boundary condition card applies. In the example data set (Figure 19), the first line of boundary conditions would apply to the first 155 model time-steps, the second line for time-step 156 through 215, etc. There must be NUM update intervals specified. The format of this card is 10I8.

278. Boundary condition cards. There is one boundary conditions card for each update interval specified (a total of NUM, specified on the update card). The format of the boundary conditions cards is identical to the first card of the initial conditions cards. Constituents that are not modeled should be left blank or set to zero.

279. Sensitivity analysis. If a sensitivity analysis is required, a title card followed by a constant card for every segment should appear behind the final boundary conditions card. The program will take on the new title and will run again with the new values specified on the constant card. A title card will be read but will not restart the program. Note also that the values do not revert. If the first run has TSIV = 0.054 and the second has TSIV = 0.11 , then the user must include in the third constant card TSIV = 0.054 if he wants it returned to its original value. Otherwise, the value retained will be the one most recently assigned, TSIV = 0.11 . More sensitivity analyses can be requested by more sets of title and constant cards, to a limit of four.

Meteorological input

280. Meteorological data, required for heat-exchange computations when ITEM is equal to zero, are provided in a separate input file (FILE2).

281. Card 1. The user here specifies the Julian day (JDYO) of the year that simulations are started. The card is arranged in two areas of eight columns each, with the first eight columns available for labeling and JDYO specified in the second area. The format for card 1 is 8X, I8.

282. Card 2. The user here specifies the dust-attenuation coefficient (DUC). Water Resources Engineers (1967) gives a range of 0.0 to 0.13 for several locations, although results are not very sensitive to changes in this range. The format for cards 2 and 3 is 8X, 6F8.2.

283. Card 3. The user here specifies the local latitude (LATUDC) and the local longitude (LONTUC) of the project, and the longitude (LSMC) of the standard meridian east of the project (standard meridians are in increments of 15 deg, e.g., 75, 90, 105, etc.). Each value on card 3 is in degrees.

284. Card 4. The user here indicates the total number (NUM) of different meteorological condition updates, right justified in the first eight columns (format I8).

285. Card 5. The user here indicates the maximum number of iterations for which the associated meteorological condition card applies. In the example input set (Figure 20), the first line of meteorological conditions would apply to the first 24 model iterations, the second line for iteration 25 through 48, etc. A total of NUM iteration numbers must be specified. The card is arranged in 10 areas of columns of eight, with right-justified integer values being specified (format 10I8).

286. Card 6. The user here specifies, respectively, the cloud cover (between 0.0 and 1.0), wind speed (miles per hour), dry bulb temperature (degrees Fahrenheit), wet bulb temperature (degrees Fahrenheit), and the atmospheric pressure (conventional inch of mercury). There should be a total of

```
       169
      0.12
     37.00    37.00    75.00
22
24      48       72       96      120      144      168      192      216      240
264    288      312      336      360      384      408      432      456      480
504    520
  1   0.80     8.00    82.00    61.00    28.81
  2   0.70     4.00    81.00    59.00    28.80
  3   0.30     5.00    81.00    60.00    28.79
  4   0.20     5.00    78.00    58.00    28.79
  5   0.20     3.00    74.00    56.00    28.80
  6   0.50     3.00    71.00    55.00    28.82
  7   0.50     3.00    65.00    54.00    28.84
  8   0.50     0.00    61.00    52.00    28.84
  9   0.50     4.00    58.00    53.00    28.85
 10   0.50     1.00    58.00    53.00    28.85
 11   0.50     0.00    54.00    50.00    28.86
 12   0.50     0.00    55.00    50.00    28.86
 13   0.50     0.00    51.00    49.00    28.86
 14   0.50     0.00    51.00    48.00    28.87
 15   0.50     4.00    53.00    49.00    28.89
 16   0.50     3.00    62.00    54.00    28.89
 17   0.60     4.00    66.00    56.00    28.90
 18   0.50     3.00    70.00    56.00    28.90
 19   0.20     3.00    73.00    59.00    28.90
 20   0.50     4.00    77.00    60.00    28.88
 21   0.50     2.00    80.00    62.00    28.87
 22   0.50     4.00    83.00    64.00    28.85
```

Figure 20. Sample meteorological data input

NUM of these cards provided. The cards are arranged in columns of eight, with the first column either left blank or used for labeling (format 8X, 5F8.0).

Output

287. The output from RIV1Q (Figure 21) consists of data for each segment at each print interval. At the top of the page appears the program title followed by the run date. The next line prints the run title supplied by the user. After that comes the elapsed time in hours and minutes, the 24-hr clock time, the date of simulation (day 1 being the first day), and the segment ID number and name. Only those parameters modeled appear in the output. Headings, units, and values appear in column format. The values of CBOD and DO are printed with one decimal place (corresponding to analytical accuracy), the remaining variables with two decimal places.

288. Graphics output from RIV1Q consists of binary data for each segment and is available with the option IPLOT. When IPLOT equals zero (the default value), no graphics output is generated; IPLOT = 10 causes only every tenth time-step to be printed to File 19. The first few lines in the graphics output consist of the river mile numbers which are written only once. The elapsed time appears on the next line with only those parameters being modeled appearing on the next few lines. The section on graphics in the source code can be located several lines below statement 505.

289. Another RIVIQ output file is File 17, which is an echo of the meteorological data that the program reads. The user can check this file to verify that the correct met values are being used.

SAMPLE DATA SET - USERS MANUAL
INITIAL CONDITIONS

	TEMP	CBODNS	ORGAN	NH3N	NO3N	PO4	DO	MN	FE	COLIFORM
1	23.40	0.01	0.01	0.10	0.01	0.02	8.50	0.00	0.00	0.00
2	22.80	0.01	0.01	0.10	0.01	0.02	8.60	0.00	0.00	0.00
3	22.80	0.01	0.01	0.10	0.01	0.02	8.70	0.00	0.00	0.00
4	24.30	0.01	0.01	0.10	0.01	0.02	8.20	0.00	0.00	0.00
5	24.40	0.01	0.01	0.10	0.01	0.02	8.20	0.00	0.00	0.00
6	24.50	0.01	0.01	0.10	0.01	0.02	8.30	0.00	0.00	0.00
7	24.70	0.01	0.01	0.10	0.01	0.02	8.30	0.00	0.00	0.00
8	25.10	0.01	0.01	0.10	0.01	0.02	8.30	0.00	0.00	0.00
9	25.40	0.01	0.01	0.10	0.01	0.02	8.30	0.00	0.00	0.00
10	25.70	0.01	0.01	0.10	0.01	0.02	8.20	0.00	0.00	0.00
11	25.90	0.01	0.01	0.10	0.01	0.02	8.20	0.00	0.00	0.00
12	26.00	0.01	0.01	0.10	0.01	0.02	8.20	0.00	0.00	0.00
13	26.00	0.01	0.01	0.10	0.01	0.02	8.20	0.00	0.00	0.00
14	25.90	0.01	0.01	0.10	0.01	0.02	8.20	0.00	0.00	0.00
15	25.70	0.01	0.01	0.10	0.01	0.02	8.30	0.00	0.00	0.00
16	25.30	0.01	0.01	0.10	0.01	0.02	8.40	0.00	0.00	0.00
17	24.80	0.01	0.01	0.10	0.01	0.02	8.50	0.00	0.00	0.00
18	24.30	0.01	0.01	0.10	0.01	0.02	8.50	0.00	0.00	0.00
19	23.90	0.01	0.01	0.10	0.01	0.02	8.60	0.00	0.00	0.00
20	23.80	0.01	0.01	0.10	0.01	0.02	8.60	0.00	0.00	0.00
21	23.70	0.01	0.01	0.10	0.01	0.02	8.60	0.00	0.00	0.00
22	23.80	0.01	0.01	0.10	0.01	0.02	8.50	0.00	0.00	0.00
23	24.00	0.01	0.01	0.10	0.01	0.02	8.50	0.00	0.00	0.00
24	25.00	0.01	0.01	0.10	0.01	0.02	8.40	0.00	0.00	0.00
25	25.60	0.01	0.01	0.10	0.01	0.02	8.30	0.00	0.00	0.00
26	26.00	0.01	0.01	0.10	0.01	0.02	8.30	0.00	0.00	0.00
27	25.40	0.01	0.01	0.10	0.01	0.02	8.30	0.00	0.00	0.00
28	24.20	0.01	0.01	0.10	0.01	0.02	8.50	0.00	0.00	0.00
29	23.40	0.01	0.01	0.10	0.01	0.02	8.50	0.00	0.00	0.00
30	23.70	0.01	0.01	0.10	0.01	0.02	8.40	0.00	0.00	0.00
31	23.90	0.01	0.01	0.10	0.01	0.02	8.20	0.00	0.00	0.00
32	23.90	0.01	0.01	0.10	0.01	0.02	8.10	0.00	0.00	0.00
33	23.90	0.01	0.01	0.10	0.01	0.02	8.10	0.00	0.00	0.00
34	23.90	0.01	0.01	0.10	0.01	0.02	8.10	0.00	0.00	0.00
35	25.00	0.01	0.01	0.10	0.01	0.02	7.20	0.00	0.00	0.00
36	25.10	0.01	0.01	0.10	0.01	0.02	7.40	0.00	0.00	0.00
37	25.40	0.01	0.01	0.10	0.01	0.02	7.50	0.00	0.00	0.00
38	26.20	0.01	0.01	0.10	0.01	0.02	7.50	0.00	0.00	0.00
39	26.30	0.01	0.01	0.10	0.01	0.02	7.60	0.00	0.00	0.00
40	26.20	0.01	0.01	0.10	0.01	0.02	7.60	0.00	0.00	0.00

Figure 21. Sample RIV1Q output (Sheet 1 of 8)

```
ITERATION UPDATES FOR BOUNDARY      1
    155      215      400      520
BOUNDARY CONDITIONS
      TEMP    CBODNS    ORGAN    NH3N     NO3N      PO4      DO       MN       FE    COLIFORM
     12.00      1.00     0.50     0.20     0.10     0.05     2.50     0.00     0.00      0.00
     15.00      1.00     0.50     0.20     0.10     0.05     4.00     0.00     0.00      0.00
     12.00      1.00     0.50     0.20     0.10     0.05     2.50     0.00     0.00      0.00
     15.00      1.00     0.50     0.20     0.10     0.05     4.00     0.00     0.00      0.00
ITERATION UPDATES FOR BOUNDARY      3
    520
BOUNDARY CONDITIONS
      TEMP    CBODNS    ORGAN    NH3N     NO3N      PO4      DO       MN       FE    COLIFORM
     20.00      1.00     0.10     0.01     0.10     0.01     8.00     0.00     0.00      0.00
$CONST
ADN     =  0.1000000    ,
AG      =   12.81000    ,
AKN     =  0.3000000    ,
AKNX    =  0.0000000E+00,
AK1     =  0.2000000    ,
ALGAEB  =  0.0000000E+00,
ATB     =  0.0000000E+00,
ATS     =  0.0000000E+00,
APO4    =  0.0000000E+00,
BK      = -0.4900000    ,
CBODL   =  0.0000000E+00,
CLABEL  =  0.00000000000000000L+00,
CSINK   =  0.0000000E+00,
CUNIT   =  0.00000000000000000E+00,
DAWN    =  0.2708334    ,
DOL     =  0.0000000E+00,
E1      =  0.5000000    ,
E2      =   1.500000    ,
IPRINT  =        192,
IPLOT   =          0,
KALGDK  =  0.1000000    ,
KALGRO  =  2.0000001E-03,
KEXT    =  0.5000000    ,
KNCBDN  =  0.1000000    ,
KOALDK  =  0.5000000    ,
KOCBDN  =  0.5000000    ,
KOCB1   =  0.5000000    ,
KON     =  0.5000000    ,
NH3NL   =  0.0000000E+00,
NO3NL   =  0 0000000E+00,
```

Figure 21. (Sheet 2 of 8)

```
ORGANL  =  0.0000000E+00,
PO4L    =  0.0000000E+00,
START   =  0.6666667    ,
SUNSET  =  0.8541666    ,
TEMP    =   23.00000    ,
TEMPL   =   25.00000    ,
TSINK   =   10.00000    ,
TSIV    =  0.0000000E+00,
KCOLIDK =   1.400000    ,
KMNDK   =  0.5000000    ,
KFEDK   =  0.5000000    ,
OXIDAT  =   1.000000    ,
MNL     =  0.0000000E+00,
FEL     =  0.0000000E+00,
COLIL   =  0.0000000E+00,
ITEM    =            0,
HNEFSW  =  0.0000000E+00,
QWINDO  = F
$END
```


Figure 21. (Sheet 3 of 8)

DYNAMIC RIVERINE WATER QUALITY MODEL,CE-QUAL-RIV1 RIV1Q 11-SEP-89
SAMPLE DATA SET - USERS MANUAL
ELAPSED TIME 0 HOURS 0 MINUTES 24-HOUR CLOCK TIME :16.00 DAY 1 SEGMENT 1 MAIN STEM ABOVE REREG DAM

	RIVER MILE	TEMP (DEG. C)	CBODNS (MG/L)	ORGAN (MG/L)	NH3N (MG/L)	NO3N (MG/L)	PO4 (MG/L)	DO (MG/L)
1	27.76	23.40	0.0	0.01	0.10	0.01	0.02	8.5
2	27.00	22.80	0.0	0.01	0.10	0.01	0.02	8.6
3	26.01	22.80	0.0	0.01	0.10	0.01	0.02	8.7
4	25.25	24.30	0.0	0.01	0.10	0.01	0.02	8.2
5	24.43	24.40	0.0	0.01	0.10	0.01	0.02	8.2
6	23.59	24.50	0.0	0.01	0.10	0.01	0.02	8.3
7	22.76	24.70	0.0	0.01	0.10	0.01	0.02	8.3
8	21.82	25.10	0.0	0.01	0.10	0.01	0.02	8.3
9	21.13	25.40	0.0	0.01	0.10	0.01	0.02	8.3
10	20.33	25.70	0.0	0.01	0.10	0.01	0.02	8.2
11	19.54	25.90	0.0	0.01	0.10	0.01	0.02	8.2
12	18.74	26.00	0.0	0.01	0.10	0.01	0.02	8.2
13	17.94	26.00	0.0	0.01	0.10	0.01	0.02	8.2
14	17.06	25.90	0.0	0.01	0.10	0.01	0.02	8.2
15	16.18	25.70	0.0	0.01	0.10	0.01	0.02	8.3
16	15.29	25.30	0.0	0.01	0.10	0.01	0.02	8.4

1DYNAMIC RIVERINE WATER QUALITY MODEL,CE-QUAL-RIV1 RIV1Q 11-SEP-89
SAMPLE DATA SET - USERS MANUAL
ELAPSED TIME 0 HOURS 0 MINUTES 24-HOUR CLOCK TIME :16.00 DAY 1 SEGMENT 2 MAIN STEM BELOW REREG DAM

	RIVER MILE	TEMP (DEG. C)	CBODNS (MG/L)	ORGAN (MG/L)	NH3N (MG/L)	NO3N (MG/L)	PO4 (MG/L)	DO (MG/L)
1	15.29	24.80	0.0	0.01	0.10	0.01	0.02	8.5
2	14.47	24.30	0.0	0.01	0.10	0.01	0.02	8.5
3	13.65	23.90	0.0	0.01	0.10	0.01	0.02	8.6
4	12.82	23.80	0.0	0.01	0.10	0.01	0.02	8.6
5	12.11	23.70	0.0	0.01	0.10	0.01	0.02	8.6
6	11.41	23.80	0.0	0.01	0.10	0.01	0.02	8.5
7	10.70	24.00	0.0	0.01	0.10	0.01	0.02	8.5
8	9.85	25.00	0.0	0.01	0.10	0.01	0.02	8.4
9	9.00	25.60	0.0	0.01	0.10	0.01	0.02	8.3
10	8.15	26.00	0.0	0.01	0.10	0.01	0.02	8.3
11	7.29	25.40	0.0	0.01	0.10	0.01	0.02	8.3
12	6.20	24.20	0.0	0.01	0.10	0.01	0.02	8.5
13	5.11	23.40	0.0	0.01	0.10	0.01	0.02	8.5
14	4.01	23.70	0.0	0.01	0.10	0.01	0.02	8.4
15	2.92	23.90	0.0	0.01	0.10	0.01	0.02	8.2
16	1.95	23.90	0.0	0.01	0.10	0.01	0.02	8.1

Figure 21. (Sheet 4 of 8)

17	0.97	23.90	0.0	0.01	0.10	0.01	0.02	8.1
18	0.00	23.90	0.0	0.01	0.10	0.01	0.02	8.1

DYNAMIC RIVERINE WATER QUALITY MODEL,CE-QUAL-RIV1 RIV1Q 11-SEP-89
SAMPLE DATA SET - USERS MANUAL
ELAPSED TIME 0 HOURS 0 MINUTES 24-HOUR CLOCK TIME :16.00 DAY 1 SEGMENT 3 TRIBUTARY UPSTRM OF REREG

	RIVER MILE	TEMP (DEG. C)	CBODNS (MG/L)	ORGAN (MG/L)	NH3N (MG/L)	NO3N (MG/L)	PO4 (MG/L)	DO (MG/L)
1	2.51	25.00	0.0	0.01	0.10	0.01	0.02	7.2
2	2.03	25.10	0.0	0.01	0.10	0.01	0.02	7.4
3	1.47	25.40	0.0	0.01	0.10	0.01	0.02	7.5
4	0.90	26.20	0.0	0.01	0.10	0.01	0.02	7.5
5	0.37	26.30	0.0	0.01	0.10	0.01	0.02	7.6
6	0.00	26.20	0.0	0.01	0.10	0.01	0.02	7.6

DYNAMIC RIVERINE WATER QUALITY MODEL,CE-QUAL-RIV1 RIV1Q 11-SEP-89
SAMPLE DATA SET - USERS MANUAL
ELAPSED TIME 24 HOURS 0 MINUTES 24-HOUR CLOCK TIME :16.00 DAY 2 SEGMENT 1 MAIN STEM ABOVE REREG DAM

	RIVER MILE	TEMP (DEG. C)	CBODNS (MG/L)	ORGAN (MG/L)	NH3N (MG/L)	NO3N (MG/L)	PO4 (MG/L)	DO (MG/L)
1	27.76	15.00	1.0	0.50	0.20	0.10	0.05	4.0
2	27.00	15.25	1.0	0.50	0.20	0.10	0.05	4.1
3	26.01	15.47	1.0	0.49	0.20	0.10	0.05	4.7
4	25.25	18.73	0.7	0.34	0.16	0.08	0.04	6.1
5	24.43	18.04	0.7	0.34	0.16	0.08	0.04	6.3
6	23.59	16.04	0.7	0.33	0.16	0.08	0.04	6.0
7	22.76	18.18	0.6	0.31	0.16	0.09	0.04	7.0
8	21.82	16.83	0.6	0.31	0.17	0.09	0.04	7.1
9	21.13	17.23	0.6	0.29	0.17	0.10	0.04	7.2
10	20.33	19.69	0.4	0.21	0.14	0.09	0.03	7.4
11	19.54	22.59	0.2	0.11	0.10	0.07	0.02	7.8
12	18.74	23.67	0.1	0.05	0.08	0.05	0.02	8.0
13	17.94	24.12	0.0	0.02	0.07	0.05	0.02	8.1
14	17.06	24.72	0.0	0.01	0.07	0.05	0.02	8.1
15	16.18	25.27	0.0	0.01	0.06	0.05	0.02	8.1
16	15.29	25.58	0.0	0.01	0.06	0.05	0.02	8.0

Figure 21. (Sheet 5 of 8)

DYNAMIC RIVERINE WATER QUALITY MODEL,CE-QUAL-RIV1 RIV1Q 11-SEP-89
SAMPLE DATA SET - USERS MANUAL
ELAPSED TIME 24 HOURS 0 MINUTES 24-HOUR CLOCK TIME :16.00 DAY 2 SEGMENT 2 MAIN STEM BELOW REREG DAM

	RIVER MILE	TEMP (DEG. C)	CBODNS (MG/L)	ORGAN (MG/L)	NH3N (MG/L)	NO3N (MG/L)	PO4 (MG/L)	DO (MG/L)
1	15.29	25.58	0.0	0.01	0.06	0.05	0.02	8.0
2	14.47	26.00	0.0	0.01	0.06	0.05	0.02	8.0
3	13.65	26.68	0.0	0.01	0.06	0.05	0.02	8.0
4	12.82	27.23	0.0	0.01	0.06	0.05	0.02	8.0
5	12.11	27.63	0.0	0.01	0.06	0.05	0.02	8.0
6	11.41	27.96	0.0	0.01	0.06	0.05	0.02	7.9
7	10.70	28.24	0.0	0.01	0.06	0.05	0.02	7.9
8	9.85	28.50	0.0	0.01	0.06	0.05	0.02	7.9
9	9.00	28.59	0.0	0.01	0.06	0.05	0.02	7.9
10	8.15	28.59	0.0	0.01	0.06	0.05	0.02	7.9
11	7.29	28.35	0.0	0.01	0.06	0.05	0.02	7.9
12	6.20	27.17	0.0	0.01	0.07	0.05	0.02	8.2
13	5.11	25.73	0.0	0.01	0.07	0.04	0.02	8.4
14	4.01	24.57	0.0	0.01	0.07	0.04	0.02	8.5
15	2.92	24.19	0.0	0.01	0.07	0.04	0.02	8.5
16	1.95	23.30	0.0	0.01	0.07	0.04	0.02	8.5
17	0.97	23.87	0.0	0.01	0.07	0.04	0.02	8.3
18	0.00	24.26	0.0	0.01	0.07	0.04	0.02	8.2

DYNAMIC RIVERINE WATER QUALITY MODEL,CE-QUAL-RIV1 RIV1Q 11-SEP-89
SAMPLE DATA SET - USERS MANUAL
ELAPSED TIME 24 HOURS 0 MINUTES 24-HOUR CLOCK TIME :16.00 DAY 2 SEGMENT 3 TRIBUTARY UPSTRM OF REREG

	RIVER MILE	TEMP (DEG. C)	CBODNS (MG/L)	ORGAN (MG/L)	NH3N (MG/L)	NO3N (MG/L)	PO4 (MG/L)	DO (MG/L)
1	2.51	24.92	0.0	0.01	0.08	0.03	0.02	8.2
2	2.03	24.95	0.0	0.01	0.08	0.03	0.02	8.2
3	1.47	25.28	0.0	0.01	0.08	0.03	0.02	8.2
4	0.90	24.97	0.0	0.01	0.08	0.03	0.02	8.2
5	0.37	24.62	0.0	0.01	0.08	0.03	0.02	8.0
6	0.00	24.58	0.0	0.01	0.07	0.04	0.02	8.0

Figure 21. (Sheet 6 of 8)

DYNAMIC RIVERINE WATER QUALITY MODEL,CE-QUAL-RIV1 RIV1Q 11-SEP-89
SAMPLE DATA SET - USERS MANUAL
ELAPSED TIME 48 HOURS 0 MINUTES 24-HOUR CLOCK TIME :16.00 DAY 3 SEGMENT 1 MAIN STEM ABOVE REREG DAM

	RIVER MILE	TEMP (DEG. C)	CBODNS (MG/L)	ORGAN (MG/L)	NH3N (MG/L)	NO3N (MG/L)	PO4 (MG/L)	DO (MG/L)
1	27.76	12.00	1.0	0.50	0.20	0.10	0.05	2.5
2	27.00	12.40	1.0	0.50	0.20	0.10	0.05	2.7
3	26.01	12.93	1.0	0.49	0.20	0.10	0.05	3.5
4	25.25	16.46	0.7	0.36	0.17	0.09	0.04	5.2
5	24.43	16.87	0.7	0.36	0.17	0.09	0.04	5.6
6	23.59	17.35	0.7	0.35	0.17	0.09	0.04	5.9
7	22.76	16.34	0.7	0.34	0.17	0.10	0.04	6.0
8	21.82	16.34	0.7	0.33	0.18	0.10	0.04	6.3
9	21.13	17.37	0.6	0.31	0.18	0.11	0.04	6.7
10	20.33	18.52	0.6	0.29	0.17	0.12	0.04	6.9
11	19.54	18.81	0.5	0.26	0.17	0.12	0.04	7.0
12	18.74	19.32	0.4	0.22	0.15	0.12	0.04	7.2
13	17.94	20.86	0.3	0.15	0.12	0.11	0.03	7.4
14	17.06	22.97	0.1	0.07	0.08	0.09	0.02	7.7
15	16.18	24.31	0.0	0.03	0.05	0.08	0.02	7.9
16	15.29	24.90	0.0	0.01	0.05	0.07	0.02	8.0

DYNAMIC RIVERINE WATER QUALITY MODEL,CE-QUAL-RIV1 RIV1Q 11-SEP-89
SAMPLE DATA SET - USERS MANUAL
ELAPSED TIME 48 HOURS 0 MINUTES 24-HOUR CLOCK TIME :16.00 DAY 3 SEGMENT 2 MAIN STEM BELOW REREG DAM

	RIVER MILE	TEMP (DEG. C)	CBODNS (MG/L)	ORGAN (MG/L)	NH3N (MG/L)	NO3N (MG/L)	PO4 (MG/L)	DO (MG/L)
1	15.29	24.90	0.0	0.01	0.05	0.07	0.02	8.0
2	14.47	25.37	0.0	0.01	0.05	0.07	0.02	8.0
3	13.65	25.98	0.0	0.01	0.04	0.07	0.02	8.0
4	12.82	26.44	0.0	0.01	0.04	0.07	0.02	8.0
5	12.11	26.81	0.0	0.01	0.04	0.07	0.02	8.0
6	11.41	27.11	0.0	0.01	0.04	0.07	0.02	8.0
7	10.70	27.35	0.0	0.01	0.04	0.07	0.02	8.0
8	9.85	27.55	0.0	0.01	0.04	0.07	0.02	8.0
9	9.00	27.62	0.0	0.01	0.04	0.07	0.02	8.0
10	8.15	27.64	0.0	0.01	0.04	0.07	0.02	8.0
11	7.29	27.47	0.0	0.01	0.04	0.07	0.02	8.0
12	6.20	26.46	0.0	0.01	0.04	0.07	0.02	8.2
13	5.11	25.39	0.0	0.01	0.04	0.07	0.02	8.4
14	4.01	24.57	0.0	0.01	0.04	0.07	0.02	8.5
15	2.92	25.03	0.0	0.01	0.04	0.07	0.02	8.3

Figure 21. (Sheet 7 of 8)

16	1.95	26.21	0.0	0.01	0.04	0.07	0.02	7.9
17	0.97	24.73	0.0	0.01	0.04	0.07	0.02	8.1
18	0.00	24.31	0.0	0.01	0.05	0.07	0.02	8.2

DYNAMIC RIVERINE WATER QUALITY MODEL,CE-QUAL-RIV1 RIV1Q 11-SEP-89
SAMPLE DATA SET - USERS MANUAL
ELAPSED TIME 48 HOURS 0 MINUTES 24-HOUR CLOCK TIME :16.00 DAY 3 SEGMENT 3 TRIBUTARY UPSTRM OF REREG

	RIVER MILE	TEMP (DEG. C)	CBODNS (MG/L)	ORGAN (MG/L)	NH3N (MG/L)	NO3N (MG/L)	PO4 (MG/L)	DO (MG/L)
1	2.51	25.06	0.0	0.01	0.07	0.04	0.02	8.2
2	2.03	25.09	0.0	0.01	0.07	0.04	0.02	8.2
3	1.47	25.42	0.0	0.01	0.07	0.04	0.02	8.2
4	0.90	25.12	0.0	0.01	0.07	0.04	0.02	8.2
5	0.37	24.82	0.0	0.01	0.07	0.05	0.02	8.2
6	0.00	24.78	0.0	0.01	0.06	0.05	0.02	8.2

Figure 21. (Sheet 8 of 8)

REFERENCES

Abbott, M. B. 1979. Computational Hydraulics, Pitman Publishing Limited, London.

Amein, M., and Chu, H. L. 1975. "Implicit Numerical Modeling of Unsteady Flows," Journal, Hydraulics Division, American Society of Civil Engineers, Vol 101, No. HY6, pp 717-731.

Amein, M., and Fang, C. S. 1970. "Implicit Flood Routing in Natural Channels," Journal, Hydraulics Division, American Society of Civil Engineers, Vol 96, No. HY12, pp 2481-2499.

Baca, R. G., and Arnett, R. C. 1976. A Limnological Model for Eutrophic Lakes and Impoundments, Report No. 2311102062, Battelle Pacific Northwest Lab, Richland, WA.

Banks, R. B. 1975. "Some Features of Wind Action on Shallow Lakes," Journal, Environmental Engineering, American Society of Civil Engineers, Division No. EE5, pp 813-827.

Bansal, M. K. 1976. "Nitrification in Natural Streams," Journal of Water Pollution Control Federation, Vol 48, pp 2380-2393.

Beck, M. B., and Young, P. C. 1975. "A Dynamic Model for BOD-DO Relationships in a Non-Tidal Stream," Water Research, Vol 9, pp 769-776.

Bennet, J. P., and Rathbun, R. E. 1972. "Reaeration in Open Channel Flow," US Geological Society Professional Paper.

Bird, R. B., Stewart, W. E., and Lightfoot, E. N. 1964. Transport Phenomena, John Wiley and Sons, New York.

Black, W. M., and Phelps, E. B. 1911. "Report Concerning the Location of Sewer Outlets and the Discharge of Sewage into New York Harbor," Contributions from the Sanitary Research Laboratory and Sewage Experiment Station, Vol VII, Massachusetts Institute of Technology, Cambridge, MA.

Bole, J. B., and Allan, J. R. 1978. "Uptake of Phosphorus from Sediment by Aquatic Plants, *Myriophyllum spicatum* and *Hydrilla verticillata*," Water Research, Vol 12, pp 353-358.

Bowles, D. S., Fread, D. and Grenney, W. 1977. "Coupled Dynamic Streamflow-Temperature Model," Journal, Environmental Engineering Division, American Society of Civil Engineers, Vol 103, No. HY5.

Brown, L. C. 1974. "Statistical Evaluation of Reaeration Prediction Equations," Journal, Environmental Engineering Division, American Society of Civil Engineers, Vol 100, pp 1051-1068.

Brown, L. C., and Barnwell, T. O. 1987. "The Enhanced Stream Water Quality Models QUAL2E and QUAL2E-UNCAS: Documentation and User Manual," EPA/600/3-87/007, US Environmental Protection Agency, Environmental Research Laboratory, Athens, GA.

Camp, T. R., and Meserve, P. L. 1974. Water and Its Impurities, 2d ed., Dowden, Hutchinson & Ross, Inc., Stroudsburg, PA.

Carignan, R., and Kalff, J. 1980. "Phosphorus Sources for Aquatic Weeds: Water or Sediments?" Science, Vol 207, pp 987-989.

Carnahan, B., Luther, H. A., and Wilkes, J. O. 1969. Applied Numerical Methods, John Wiley and Sons, New York.

Churchill, M. A., Elmore, H. L., and Buckingham, R. A. 1962. "The Prediction of Stream Reaeration Rates," International Journal of Air and Water Pollution, Vol 6, pp 467-504.

Covar, A. P. 1976. "Selecting the Proper Reaeration Coefficient for Use in Water Quality Models," Proceedings of the Conference on Environmental Modelling and Simulation, Cincinnati, US Environmental Protection Agency, Cincinnati, pp 340-343.

Curtis, E. J. C., Durrant, K., and Harman, M. M. I. 1975. "Nitrification in Rivers in the Trent Basin," Water Research, Vol 9, pp 225-268.

DiToro, D. M., Thomann, R. V., O'Connor, D. J., and Mancini, J. L. 1977. "Estuarine Phytoplankton Biomass Models--Verification Analyses and Preliminary Applications," The Sea, Vol 6, E. D. Goldberg et al., ed., John Wiley and Sons, New York, pp 969-1020.

Donigan, A. S., and Crawford, N. H. 1979. "Water Quality Model for Agricultural Runoff," Modeling of Rivers, H. W. Shen, ed., John Wiley and Sons, New York, pp 12-1 - 12-82.

Edinger, J. E., Brady, D. K., and Geyer, J. C. 1974. "Heat Exchange and Transport in the Environment," Report No. 14, EPRI Publication No. 74-049-00-3, prepared for the Electric Power Research Institute, Palo Alto, CA.

Edinger, J. E., Duttweiler, D., and Geyer, J. 1968. "The Response of Water Temperatures to Meteorological Conditions," Water Resources Research, Vol 4, No. 5.

Elmore, H. C., and West, W. F. 1961. "Effect of Water Temperature on Stream Reaeration," Journal, Sanitation Engineering Division, American Society of Civil Engineers, Vol 87, No. SA6, pp 59-71.

Elmore, H. L., and Hayes, T. W. 1960. "Solubility of Atmospheric Oxygen in Water," Journal, Sanitation Engineering Division, American Society of Civil Engineers, Vol 86, No. SA4, pp 41-53.

Engelbrecht, R. S., and Morgan, J. J. 1959. "Studies on the Occurrence and Degradation of Condensed Phosphate in Surface Waters," Sewage and Industrial Wastes, Vol 31, pp 458-478.

Fischer, H. B., List, E. J., Koh, R. C., Imberger, J., and Brooks, N. H. 1979, Mixing in Inland and Coastal Waters, Academic Press, New York.

Fitzgerald, G. P. 1970. "Aerobic Lake Muds for the Removal of Phosphorus from Lake Waters," Limnology and Oceanography, Vol 15, pp 550-555.

Ford, D. E., and Johnson, L. S. 1986. "An Assessment Reservoir Mixing Processes," Technical Report E-86-7, prepared by Ford, Thornton, Norton and Associates, Ltd., Little Rock, AR, for the US Army Engineer Waterways Experiment Station, Vicksburg, MS.

Foree, E. G., and McCarty, P. L. 1968. The Decomposition of Algae in Anaerobic Waters, Technical Report No. 95, Department of Civil Engineering, Stanford University, Stanford, CA.

Fread, D. L. 1973. "Technique for Implicit Dynamic Routing in Rivers with Tributaries," Water Resources Research, Vol 9, No. 4, pp 918-925.

_____. 1978. DAMBRK: The NWS Dam-Break Flood Forecasting Model, Office of Hydrology, National Weather Service, Silver Spring, MD.

Fread, D. L., and Smith, C. F. 1978. "Calibration Technique for 1-D Unsteady Flow Models," Journal, Hydraulics Division, American Society of Civil Engineers, Vol 104, No. HY7, pp 1027-1044.

Frost, W. H., and Streeter, H. W. 1924. Public Health Bulletin, No. 143, US Public Health Service, Washington, DC.

Garland, J. H. N. 1978. "Nitrification in the River Trent," Mathematical Models in Water Pollution Control, A. James, ed., John Wiley and Sons, New York.

Garland, J. H. N., and Rolley, H. L. J. 1977. "Studies of River Water Quality in the Lancashire Tame," Water Pollution Control, Vol 76, pp 301-326.

Gerloff, G. C. 1969. "Evaluating Nutrient Supplies for the Growth of Aquatic Plants in Natural Waters," Eutrophication: Causes, Consequences, Correctives, National Academy of Sciences, Washington, DC, pp 537-555.

Gummerman, R. C. 1970. "Aqueous Phosphate and Lake Sediment Interaction," Proceedings of the 13th Conference on the Great Lakes Reservoir, Vol 13, pp 675-683.

Haslam, S. M. 1978. River Plants, Cambridge University Press, Cambridge, MA.

Hill, A. R., and Sanmugadas, K. 1985. "Denitrification Rates in Relation to Stream Sediment Characteristics," Water Research, Vol 19, No. 12, pp 1579-1586.

Holler, A. G. 1970. "Reaeration of Discharge Through Hydraulic Structures," US Army Engineer Division, Ohio River, Cincinnati, OH.

Holly, F. M., and Cunge, J. 1979. "Coupled Dynamic Streamflow - Temperature Models Discussion," Journal, Hydraulics Division, American Society of Civil Engineers, Vol 104, pp 316-318.

Holly, F. M., and Preissman, A. 1978. "Accurate Calculation of Transport in Two Dimensions," Journal, Hydraulics Division, American Society of Civil Engineers, Vol 103, No. HY11, pp 1259-1277.

Hoover, S. R., and Porges, N. 1952. "Assimilation of Dairy Wastes by Activated Sludge--II. The Equation of Synthesis and Rate of Oxygen Utilization," Sewage and Industrial Wastes, Vol 24, pp 306-312.

Hynes, H. B. N. 1970. The Ecology of Running Waters, University of Toronto Press, Toronto.

International Business Machines Corp. 1968. "IBM System/360 Scientific Subroutine Package," 360A-CM-03X Version III Programmer's Manual.

_____. 1974. "IBM System/360 and System/370 FORTRAN IV Language," CG28-6515-10.

Jewell, W. J., and McCarty, P. O. 1968. Aerobic Decomposition of Algae and Nutrient Regeneration, Technical Report 91, Department of Civil Engineering, Stanford University, Stanford, CA.

Kittrell, F. W., and Furfari, S. A. 1963. "Observations of Coliform Bacteria in Streams," Journal of Water Pollution Control Federation, Vol 35, pp 1361-1385.

Knuth, D. E. 1973. "Sorting and Searching," The Art of Computer Programming, Vol 3, Addison-Wesley.

Krombach, H., and Barthel, J. 1963. "Investigation of a Small Watercourse Accidentally Polluted by Phenol Compounds," International Journal of Air and Water Pollution, Vol 7, pp 39-46.

Leopold, L. B., Wolman, M. G., and Miller, J. P. 1964. Fluvial Processes in Geomorphology, W. H. Freeman and Co., San Francisco, CA.

Liggett, J. A. 1975. "Basic Equations of Unsteady Flow," Chapter 2, Unsteady Flow in Open Channels, K. Mahmood and V. Yevjevitch, eds., Water Resources Publication, Fort Collins, CO.

Liggett, J. A., and Cunge, J. A. 1975. "Numerical Methods of Solution of the Unsteady Flow Equations," Chapter 4, Unsteady Flow in Open Channels, K. Mahmood and V. Yevjevitch, eds., Water Resources Publication, Fort Collins, CO.

Liggett, J. A., and Woolhiser, D. A. 1967. "Difference Solutions of the Shallow Water Equation," Journal, Engineering Mechanics Division, American Society of Civil Engineers, Vol 93, No. EM2, pp 39-71.

Lyn, D. A., and Goodwin, P. 1987. "Stability of a General Preissman Scheme," Journal, Hydraulic Engineering, American Society of Civil Engineer, Vol 113, No. 1, pp 16-28.

MacKenthum, K. M., and Ingram, W. M. 1967. Biological Associated Problems in Freshwater Environments, Federal Water Pollution Control Administration, US Department of the Interior, Washington, DC.

Matter, W., Hudson, P., Nestler, J., and Saul, G. 1983. "Movement, Transport, and Scour of Particulate Organic Matter and Aquatic Invertebrates Downstream from a Peaking Hydropower Project," Technical Report E-83-12, US Army Engineer Waterways Experiment Station, Vicksburg, MS.

Miller, J. E., and Jennings, M. E. 1979. "Modeling Nitrogen, Oxygen, Chattahoochee River, Ga.," Journal, Environmental Engineering Division, American Society of Civil Engineers, Vol 105, pp 641-653.

Novotny, V. 1969. "Boundary Layer Effect on the Course of the Self-Purification of Small Streams," Advances in Water Pollution Research, S. H. Jenkins, ed., Pergamon Press, Oxford, pp 39-55.

O'Connor, D. J. 1983. "Wind Effects on Gas-Liquid Transfer Coefficients," Journal, Environmental Engineering Division, American Society of Civil Engineers, Vol 109, pp 731-752.

O'Connor, D. J., and Dobbins, W. E. 1958. "The Mechanism of Reaeration in Natural Streams," Journal, Sanitary Engineering Division, American Society of Civil Engineers, Vol 82, pp 1-30.

O'Connor, D. J., Thomann, R. V., and DiToro, D. M. 1973. "Dynamic Water Quality Forecasting and Management," EPA-660/3-73-009, Office of Research and Development, US Environmental Protection Agency, Washington, DC.

Olsen, S. 1964. "Phosphate Equilibrium between Reduced Sediments and Water," International Association of Theoretical Applied Limnology, Vol 15, pp 333-341.

Owens, M., Edwards, R. W., and Gibbs, J. W. 1964. "Some Reaeration Studies in Streams," International Journal of Air and Water Pollution, Vol 8, pp 469-486.

Parker, D. S., Stone, R. W., Stenquist, R. J., and Culp, G. 1975. Process Design Manual for Nitrogen Control, Office of Technology Transfer, US Environmental Protection Agency, Washington, DC.

Preissmann, A. 1961. "Propagation des Intumescences dans les Canaux et Rivieres," First Congress of the Calculation Society of France, Grenoble, pp 433-442.

Roesner, L. A., Giguere, P. R., and Evenson, D. E. 1981. "Computer Program Documentation for the Stream Quality Model QUAL-II," EPA 600/9-81-014, Water Resources Engineers, Inc., Walnut Creek, CA.

Rogers, D., and Adams, J. 1976. Mathematical Elements for Computer Graphics, McGraw-Hill, New York.

Ruane, R. J., and Krenkel, P. A. 1977. "Nitrification and Other Factors Affecting Nitrogen in the Holston River," Progress in Water Technology, Vol 8, No. 4-5, pp 209-224.

Schroepfer, G. J., Robins, M. L., and Susag, R. H. 1960. "A Reappraisal of Deoxygenation Rates of Raw Sewage Effluents and Receiving Waters," Journal of Water Pollution Control Federation, Vol 32, pp 1212-1231.

Strand, S. 1975. Statistics of the BOD Parameters, M.S. thesis, Ohio State University, Columbus, OH.

Streeter, H. W., and Phelps, E. B. 1925. "A Study of the Pollution and Natural Purification of the Ohio River--III. Factors Concerned in the Phenomena of Oxidation and Reaeration," Bulletin No. 146, US Public Health Service, Cincinnati, OH.

Streeter, V., and Wylie, E. B. 1979. Fluid Mechanics, McGraw-Hill, New York.

Taylor, A. W., and Kunishi, H. M. 1971. "Phosphate Equilibria on Stream Sediment and Soil in a Watershed Draining and Agricultural Region," Journal of Agriculture Food Chemistry, Vol 19, pp 827-831.

Theriault, E. J. 1927. The Oxygen Demand of Polluted Waters, Bulletin No. 173, US Public Health Service, Washington, DC.

Tsivoglou, E. C., and Wallace, J. R. 1972. Characterization of Stream Reaeration Capacity, EPA-R3-72-012, Ecology Research Service, Office of Research and Monitor, US Environmental Protection Agency, Washington, DC.

Velz, C. J. 1939. "Deoxygenation and Reoxygenation," Transactions of the American Society of Civil Engineers, Vol 104, pp 560-578.

_____. 1947. "Factors Influencing Self-Purification and Their Relation to Pollution Abatement," Sewage Works Journal, Vol 19, pp 629-644.

_____. 1970. Applied Stream Sanitation, John Wiley and Sons, New York.

Water Resources Engineers, Inc. 1967. "Prediction of Thermal Energy Distribution in Streams and Reservoirs," prepared for the California Department of Fish and Game.

Wheatland, A. B., Barrett, M. J., and Bruce, A. M. 1959. "Some Observations on Denitrification in Rivers and Estuaries," Journal and Proceedings of the Institute of Sewage Purification, pp 149-162.

Whitton, B. A., ed. 1975. River Ecology, University California Press, Berkeley, CA.

Wilhelms, S. C., and Smith, D. R. 1981. "Reaeration Through Gated-Conduit Outlet Works," Technical Report E-81-5, US Army Engineer Waterways Experiment Station, Vicksburg, MS.

Williams, R. E., and Lewis, M. S. 1986. "Stream Model of Benthic Nitrification-Denitrification," Journal, Environmental Engineering, American Society of Civil Engineers, Vol 112, No. 2, pp 367-386.

Wilson, G. T., and Macleod, N. 1974. "A Critical Appraisal of Empirical Equations and Models for the Prediction of the Coefficient of Reaeration of Deoxygenated Water," Water Research, Vol 8, pp 341-366.

Wright, R. M., and McDonnell, A. J. 1979. "In-Stream Deoxygenation Rate Prediction," Journal, Environmental Engineering Division, American Society of Civil Engineers, Vol 105, pp 323-335.

Yalin, M. S. 1977. Mechanics of Sediment Transport, 2d ed., Pergamon Press, Oxford.

Yen, B. C. 1979. "Unsteady Flow Mathematical Modeling Techniques," Modeling of Rivers, H. W. Shen, ed. John Wiley and Sons, New York.

Zison, S. W., Mills, W. B., Deimer, D., and Chen, C. W. 1978. Rates, Constants, and Kinetics Formulations in Surface Water Quality Modeling, EPA-600/3-78-105, Environmental Research Laboratory, US Environmental Protection Agency, Athens, GA.

APPENDIX A: NON-AMERICAN NATIONAL STANDARD INSTITUTE LANGUAGE FEATURES

1. The CE-QUAL-RIV1 codes make use of several non-American National Standard Institute (ANSI) FORTRAN features of the FORTRAN IV language. Some of these extensions to the language are essential to the organization of the program; others merely facilitated writing and understanding of the program. Each feature is discussed in the order it appears in International Business Machines Corp. (IBM) (1968, 1974).*

2. Character strings, employed in input and output operations, are assigned values by enclosing the characters in apostrophes rather than through Hollerith literals. These strings appear in FORMAT and DATA statements as well as where initial values are assigned in type-specification statements.

3. In the water quality program with all the K-rates used, it is convenient to declare all variables beginning with the letter "K" as real rather than integer. This is done by the nonstandard IMPLICIT statement. It appears as IMPLICIT REAL (K). All other letters retain their default status (A-H and O-Z are real; I, J, L, M, and N are integer).

4. The END=n parameter on the READ statement causes the program to go to statement n when an end-of-file is encountered in attempting to read. In the hydrodynamic program, this is used to stop execution upon exhaustion of input data. In the water quality program, it is used to test when no more sensitivity analyses are requested.

5. Constants can be supplied by the user to override default values (or in the case of the water quality program, values used in previous runs) by NAMELIST input. In this form of input, variables (which must appear in the program's NAMELIST dictionary) are assigned values by name in the input deck (see Input and Output, paragraphs 216 and 269-270 in the main text). Variables that do not appear in the input retain their most recently assigned values.

6. The "Tr" format code indicates that the next format item should begin in column r. This code is used, for instance, where the time of day is printed with a colon. As an example, the clock time is stored in the program as a fraction of a day. A value of 0.4 would be mathematically converted for output to HH.MM format for 9.36. In the FORMAT statement, r is selected such that Tr, ':' will print a colon in place of the period (9:36).

---

* See References at the end of the main text.

7. Two extensions to the explicit type specification statement are employed. One is length specification (REAL*8 is used instead of DOUBLE PRECISION) and the other is assignment of initial values (in ANSI FORTRAN this can take place only in a DATA statement).

8. List-directed input allows input values to appear anywhere on the card as long as they are in order and are separated by a comma or blanks or both. It is not a standard feature of the FORTRAN IV language, but it is included in most compilers. It is indicated in the program as READ (5,*).

9. DATE is not a language feature at all, but an in-house routine. Most installations, however, should have something similar that can supply to the program the current date (or it can be suppressed entirely without affecting the calculations).

APPENDIX B: LISTING OF THE HYDRODYNAMIC CODE RIV1H

```
C  ****************. ************************************************
C  *                                                                *
C  *                    CE-QUAL-RIV1    VAX VERSION                 *
C  *                  RIV1H CODE - HYDRODYNAMIC MODEL               *
C  *                                                                *
C  *  THIS VERSION OF CE-QUAL-RIV1H CORRESPONDS TO THE FIRST        *
C  *  RELEASE VERSION OF THE USER MANUAL.                           *
C  *  THIS MODEL WAS ORIGINALLY DEVELOPED BY OHIO STATE U.,         *
C  *     BEDFORD, SYKES, AND LIBICKI.                               *
C  *  THIS VERSION HAS BEEN MODIFIED BY:                            *
C  *    M. DORTCH AND T. SCHNEIDER OF THE                           *
C  *    WATER QUALITY MODELING GROUP                                *
C  *    USAE WATERWAYS EXPERIMENT STATION                           *
C  *    VICKSBURG, MISSISSIPPI, AND                                 *
C  *    D. GRIFFIN, JR.                                             *
C  *    LOUISIANA TECH UNIVERSITY                                   *
C  *    RUSTON, LOUISIANA                                           *
C  ****************************************. ************************
C
C
C                 PROGRAM FOR CALCULATING TIME-VARYING
C                 CROSS-SECTIONAL AREAS AND FLOWS AT
C                 VARIOUS STREAM SECTIONS DURING EPISODES
C
C                 GOVERNING EQUATIONS ARE THE CONSERVATION
C                 OF MASS AND THE CONSERVATION OF MOMENTUM
C                 WHICH ARE SOLVED BY THE FOUR-POINT
C                 IMPLICIT FINITE DIFFERENCE METHOD.
C
C
C  ***************************************************************************
C
C     MODEL RESULTS ARE WRITTEN TO FILE 6 AND FILE 8.
C     DIAGNOSTICS ARE WRITTEN TO FILE 7.
C
      PROGRAM RIV1H

      COMMON DJ(70000),Z(500),XMAN(500),DNDH(500),AX(500)
      COMMON/ABLOCK/IND1,IND2,IND3,IND4,MTIME,NS,NS1,TITLE
      COMMON/MNODE/MNODE
      DIMENSION TITLE(20)

      OPEN (UNIT=5,FILE='FILE1',STATUS='OLD')
      OPEN (UNIT=6,FILE='FILE2',STATUS='NEW')

      READ(5,10)TITLE
      READ(5,*)MNODE,MTIME,NS
C
      IND1=MNODE
      IND2=10*(MNODE-1)
      IND3=2*MNODE
      IND4=4*MNODE
      NS1 = NS+1
      N1 = IND1+1
```

```
      N2 = N1+IND1
      N3 = N2+IND1
      N4 = N3+IND1
      N5 = N4+IND1
      N6 = N5+IND1
      N7 = N6+IND1
      N8 = N7+IND1
      N9 = N8+IND1
      N10 = N9+IND1
      N11 = N10+IND1
      N12 = N11+IND1
      N13 = N12+IND1
      N14 = N13+IND1
      N15 = N14+IND1
      N16 = N15+IND1
      N17 = N16+IND3
      N18 = N17+IND3
      N19 = N18+NS
      N20 = N19+NS
      N21 = N20+NS
      N22 = N21+NS
      N23 = N22+NS
      N24 = N23+NS
      N25 = N24+NS
      N26 = N25+NS
      N27 = N26+NS
      N28 = N27+NS
      N29 = N28+NS
      N30 = N29+NS
      N31 = N30+NS
      N32 = N31+NS
      N33 = N32+NS
      N34 = N33+NS
      N35 = N34+4*NS
      N36 = N35+10*NS
      N37 = N36+NS1
      N38 = N37+NS1
      CALL MAIN2(DJ(1),DJ(N1),DJ(N2),DJ(N3),DJ(N4),DJ(N5),DJ(N6),DJ(N7),
     @DJ(N8),DJ(N9),DJ(N10),DJ(N11),DJ(N12),DJ(N13),DJ(N14),DJ(N15),
     @DJ(N16),DJ(N17),DJ(N18),DJ(N19),DJ(N20),DJ(N21),DJ(N22),DJ(N23),
     @DJ(N24),DJ(N25),DJ(N26),DJ(N27),DJ(N28),DJ(N29),DJ(N30),DJ(N31),
     @DJ(N32),DJ(N33),DJ(N34),DJ(N35),DJ(N36),DJ(N37),DJ(N38),DJ(1),
     @Z(IND1),XMAN(IND1),DNDH(IND1),AX(IND1))
      STOP
   10 FORMAT(20A4)
      END
C
C
C  ***********************************************************************
C
C     THIS IS THE MAIN PROGRAM CONTAINING THE READ, WRITE, AND CONTROL
C     PORTIONS OF THE MAIN TIME MARCH (WHICH ENDS WHEN INPUT DATA IS
C     EXHAUSTED).  THE NONLINEAR EQUATIONS ARE SOLVED SIMULTANEOUSLY
C     BY THE NEWTON-RAPHSON ITERATIVE PROCEDURE.
C
```

```
C  ***********************************************************************
C
      SUBROUTINE MAIN2(Q,A,B,EL,DX1,QL,H,I3,C1,C2,C3,CN1,KE1,XC,XM,RMILE
     @,C,R,LIB,ID,NODE1,NNODE,FEEDS,ORDER,JNODE,JBCU,JBCD,BCU,BCD,JT,
     @COSP,IT0,IT1,IT2,T,SNAME,IBC,BC,AA,HYDRO,Z,XMAN,DNDH,AX)
      COMMON/ABLOCK/IND1,IND2,IND3,IND4,MTIME,NS,NS1,TITLE
      COMMON/MNODE/MNODE
      REAL KE1
      INTEGER ORDER,FEEDS
      DIMENSION TITLE(20),DX1(IND1),Q(IND1),A(IND1),B(IND1),I3(IND1),
     @C1(IND1),C2(IND1),C3(IND1),H(IND1),EL(IND1),QL(IND1),CN1(IND1),
     @KE1(IND1),R(IND3),AA(IND2),XC(IND1),XM(IND1),RMILE(IND1),
     @HYDRO(IND4),C(IND3),LIB(NS),ID(NS),NODE1(NS),
     @NNODE(NS),FEEDS(NS),ORDER(NS),JNODE(NS),JBCU(NS),JBCD(NS),
     @BCU(NS),BCD(NS),JT(NS),COSP(NS),IT0(NS),IT1(NS),IT2(NS),
     @T(4,NS),SNAME(10,NS),IBC(NS1),BC(NS1),Z(IND1),XMAN(IND1),
     @DNDH(IND1),AX(IND1)
C ***********************************************************************
C      200 UPDATE INTERVALS, 10 BOUNDARY CONDITIONS
C ***********************************************************************
       DIMENSION IUPD(200),DUMMY(10,200),DLT(200)
C
      CHARACTER*9 ADATE
      DATA GR/32.174/,THETA/0.55/,TOLER/0.001/,BETA/1.0/,FLOW/'Q'/,
     @RCURVE/'R'/,IPRINT/1/
      NAMELIST/CONST/BETA,GR,IPRINT,RMILE0,THETA,TOLER
      READ(5,CONST)
      M2 = 0
      DO 1 L=1,NS
       NODE1(L) = M2+1
C
C  READ THE SEGMENT CARD
C
       READ(5,2)ID(L),(SNAME(J,L),J=1,10),NNODE(L),FEEDS(L),JNODE(L),
     @ BTU,BCU(L),BTD,BCD(L),COSP(L)
    2 FORMAT(I2,10A4,3I3,2(A1,F8.0),F8.0)
C
C  PARSE THE SEGMENT CARD
C
       LIB(L)=1
       IF(BTU.EQ.FLOW)LIB(L) = 4
       IF(BTD.EQ.FLOW)LIB(L) = LIB(L)+1
       IF(BTD.EQ.RCURVE)LIB(L) = LIB(L)+2
C
C  CHECK TO SEE IF THIS SEGMENT IS A TRIBUTARY
C
       IF(JNODE(L).EQ.0)GO TO 4
       COSP(L) = COS(0.0174533*COSP(L))
       IF(LIB(L).LE.3)GO TO 5
       LIB(L) = 4
       GO TO 4
    5  LIB(L) = 1
    4  CONTINUE
       M1 = NODE1(L)
       M2 = M1+NNODE(L)-1
```

```
C
C  READ INITIAL CONDITIONS CARDS
C
       DO 7 I=M1,M2
       READ(5,20)DX1(I),Q(I),H(I),QL(I),EL(I),C1(I),C2(I),C3(I),
     @ CN1(I),KE1(I),AX(I),DNDH(I)
    7 CONTINUE
    1 CONTINUE
      WRITE(6,30)TITLE
      WRITE(6,CONST)
      WRITE(6,40)(DX1(I),Q(I),H(I),QL(I),EL(I),C1(I),C2(I),C3(I),
     @            CN1(I),KE1(I),AX(I),DNDH(I),I=1,IND1)
C
C  READ THE BOUNDARY CONDITIONS ID CARD
C
      DO 26 L=1,NS1
   26  IBC(L) = 0
      READ(5,*)IBC
      WRITE(6,*)IBC
C
C  UNSCRAMBLE REFERENCES
C
      DO 31 L=1,NS
       JBCU(L) = 0
       JBCD(L) = 0
       DO 31 M=1,NS
        IF(FEEDS(L).EQ.ID(M))FEEDS(L) = M
   31 CONTINUE
      NBC = 0
      DO 29 L=1,NS1
       IF(IBC(L).EQ.0)GO TO 24
       NBC =NBC+1
       DO 25 M=1,NS
        IF(IBC(L).EQ.ID(M))IBC(L) = M
        IF(IBC(L).EQ.-ID(M))IBC(L) = -M
   25  CONTINUE
   29 CONTINUE
   24 CONTINUE
C
C INPUT FOR BOUNDARY CONDITIONS
C NUM= NUMBER OF UPDATES, UPD=UPDATE ITERATIONS
C
        READ(5,101)NUM
        WRITE(6,101)NUM
        READ(5,101)(IUPD(LLL),LLL=1,NUM)
        WRITE(6,101)(IUPD(LLL),LLL=1,NUM)
          DO 201 LLL=1,NUM
              READ(5,150)DLT(LLL),(DUMMY(III,LLL),III=1,NBC)
             WRITE(6,150)DLT(LLL),(DUMMY(III,LLL),III=1,NBC)
201       CONTINUE
101     FORMAT(7I10)
          ICOUNT=1
C
C  CONSTRUCT CROSS-REFERENCE BOUNDARY CONDITIONS DIRECTORY
C
```

```
      IF(NBC.LE.0)GO TO 33
      DO 32 L=1,NBC
       IF(IBC(L))28,32,27
   28  JBCD(-IBC(L)) = L
       GO TO 32
   27  JBCU(IBC(L)) = L
   32 CONTINUE
   33 CONTINUE
C
C  ESTABLISH AN "UPSTREAM" ORDERING OF THE SEGMENTS
C
      DO 42 I=1,NS
       ORDER(I) = 0
       IT1(I) = 0
       IT0(I) = 0
       IT2(L) = 0
       IF(FEEDS(I).NE.0)GO TO 42
       ORDER(1) = I
   42 CONTINUE
      L = 1
      IT = 0
      DO 44 I=1,NS
       M = ORDER(I)
       DO 44 J=1,NS
        IF(FEEDS(J).NE.M)GO TO 44
        L = L+1
        ORDER(L) = J
        IF(JNODE(J).GT.0)GO TO 39
        JBCU(M) = -(NODE1(J)+NNODE(J)-1)
        IF(LIB(M).LE.3)LIB(M) = LIB(M)+3
        GO TO 38
   39   IT = IT+1
        IF(IT1(M).EQ.0)IT1(M) = IT
        JT(IT) = JNODE(J)
        IT2(M) = IT
        IT0(J) = IT
        JBCD(J) = -(NODE1(M)+JNODE(J)-1)
   38   IF(L.EQ.NS)GO TO 43
   44  CONTINUE
   43 CONTINUE
      IF(NS.GT.2)CALL BUBBLE(IT0,IT1,IT2,JT,NS)
C
C  TRANSFORM INITIAL STAGES INTO INITIAL CROSS-SECTIONAL AREAS AND
C  CHANNEL TOP-WIDTHS.  C3<0 IS THE SIGNAL THAT AN ELLIPSOID CROSS-
C  SECTION IS INDICATED.
C
      DO 70 I=1,IND1
       XMAN(I)=CN1(I)
       CN1(I) = CN1(I)**2/2.2*GR
       Z(I) = EL(I)
       EL(I) = H(I)+EL(I)
       IF(C3(I).LT.0.0) GO TO 60
       IF(AINT(C3(I)).EQ.C3(I))GO TO 65
       I3(I) = 0
       A(I)=C1(I)*H(I)+C2(I)*H(I)**C3(I)
```

```
      B(I)=C1(I)+C2(I)*C3(I)*H(I)**(C3(I)-1.)
      GO TO 70
   65 I3(I) = C3(I)
      A(I)=C1(I)*H(I)+C2(I)*H(I)**I3(I)
      B(I)=C1(I)+C2(I)*C3(I)*H(I)**(I3(I)-1)
      GO TO 70
   60 I3(I) = -1
      B(I)=2.*C2(I)/C1(I)*SQRT((2.*C1(I)-H(I))*H(I))
      A(I)=C1(I)*C2(I)*ACOS(1.-H(I)/C1(I))-B(I)*(C1(I)-H(I))/2.
   70 CONTINUE
C
C  CALCULATE RIVERMILE FOR EACH POINT IN THE NETWORK
C
      DO 49 LL=1,NS
       L = ORDER(LL)
       M1 = NODE1(L)+1
       M2 = NNODE(L)+M1-2
       IF(LL.GT.1)GO TO 51
       RMILE(M2) = RMILE0
       GO TO 52
   51  IF(JNODE(L).GT.0)GO TO 53
       LJ = FEEDS(L)
       MJ = NODE1(LJ)
       RMILE(M2) = RMILE(MJ)
       GO TO 52
   53  RMILE(M2) = 0.
   52  CONTINUE
       DO 54 II=M1,M2
        I = M1+M2-II
        RMILE(I-1) = RMILE(I)+DX1(I-1)/5280.
   54  CONTINUE
   49  CONTINUE
C
C  WRITE OUT INITIAL DATA
C
      WRITE(8)IND1,MTIME,NS
      WRITE(8)JT,IT0,IT1,IT2,ORDER,NNODE,NODE1,ID,JBCU
      WRITE(8)DX1,QL,RMILE
      G2 = GR/2.
      CALL DATE(ADATE)
C
        ITIME = 0
        ELAPSE = 0.0
   80 CONTINUE

C  CALCULATE ROOT MEAN SQUARE Q & H FOR TOLERANCE TEST
C
      RMSQ = 0.
      RMSA = 0.
      DO 66 I=1,IND1
       RMSQ = RMSQ+Q(I)*Q(I)
       RMSA = RMSA+A(I)*A(I)
   66 CONTINUE
      RMSQ = TOLER*SQRT(RMSQ)/FLOAT(IND1)
      RMSA = TOLER*SQRT(RMSA)/FLOAT(IND1)
```

```
C
C  BEGIN THE MAIN TIME MARCH LOOP
C
C  WRITE OUT THE PREVIOUS TIME STEP'S DATA
C
       IEHOUR=ELAPSE*24.+0.008
       IEMIN=(ELAPSE*24.-IEHOUR)*60.+0.5
      IF(MOD(ITIME,IPRINT).NE.0)GO TO 82
      DO 81 L=1,NS
       WRITE(6,100)ADATE
       WRITE(6,110)TITLE
       WRITE(6,120)IEHOUR,IEMIN,ITIME,ID(L),(SNAME(J,L),J=1,10)
      WRITE(6,130)
      M = NODE1(L)-1
      M2 = NNODE(L)

      WRITE(6,140)(I,RMILE(I+M),Q(I+M),A(I+M),B(I+M),H(I+M),EL(I+M),
     @ XMAN(I+M),I=1,M2)

   81 CONTINUE
   82 CONTINUE
      WRITE(8)DT,HYDRO
C
C ***************************************************************************
C *CHECK COURANT NO.  IF GREATER THAN 1.0, PRINT TO FILE 7.                *
C *COURANT NO. GREATER THAN 1.0 WILL CAUSE THE RIV1Q CODE TO GO UNSTABLE.*
C ***************************************************************************
C
         MNM1 = MNODE - 1
         DO 683 IK = 2, MNM1
           COURANT = ( Q(IK) * DT ) / ( DX1(IK) * A(IK) )
           IF ( COURANT .GT. .99 ) WRITE(7,684) IK, ELAPSE, COURANT
 683     CONTINUE
 684     FORMAT (10X, 'NODE    ', I3, 10X, 'ELAPSE = ', F10.6,
     @   5X, 'COURANT NO = ', F6.2 )
 685     FORMAT (///,10X,'NODE     ', I3, 10X, 'ELAPSE = ',F10.6, 'FLOW
     @   IS NEGATIVE')
C
      ITIME=ITIME+1
C
C  UPDATE TIMES
      IF(ICOUNT.GT.NUM)RETURN

      DT=DLT(ICOUNT)
      DO 301 J=1,NBC
         BC(J) = DUMMY(J,ICOUNT)
301   CONTINUE

      ELAPSE=ELAPSE+DT/86400.

      IF(ITIME.GE.IUPD(ICOUNT))ICOUNT=ICOUNT+1
C ***************************************************************************
C                                                                            *
C   VARIABLE MANNING'S N VALUES:                                             *
C   MANNING'S n IS ALLOWED TO VARY WITH DEPTH USING A LINEAR RELATIONSHIP    *
```

```
C   WITH AX (THE INTERCEPT) AND DNDH (THE PARTIAL OF N WITH RESPECT TO     *
C   HEIGHT)                                                                *
C   THE "DO 165" LOOP BELOW COMPUTES THE VALUES OF THE VARIABLE MANNING'S  *
C   n, XMAN, AND THE VALUE OF CN1(I) FOR EACH NODE FOR THE FIRST ITERATION *
C   WITHIN A TIME STEP.  IF ADDITIONAL ITERATIONS WITHIN THE SAME TIME     *
C   STEP ARE REQUIRED, A SIMILAR LOOP IS PROVIDED JUST PRIOR TO            *
C   RE-ENTERING SUBROUTINE ITER BELOW.                                     *
C ************************************************************************
      DO 165 I=1,IND1
        IF(AX(I).NE.0.0.AND.DNDH(I).NE.0.0)THEN
            XMAN(I)=AX(I) - DNDH(I)*H(I)
          IF(XMAN(I).LE.0.01)THEN
            WRITE(7,177)I
  177     FORMAT('VALUE FOR XMAN AT I =',I9,' IS  < 0.01- SET TO 0.01')
             XMAN(I)=0.01
          END IF
         CN1(I)=XMAN(I)**2*GR/2.2
        END IF
  165 CONTINUE


C  CALLING SUBROUTINES
C
      DO 400 LL=1,NS
       L = ORDER(NS-LL+1)
       M = NODE1(L)
       M1 = NNODE(L)
       M2 = 2*M-1
       IF(JBCU(L))401,402,403
  401  JJ = -JBCU(L)
       BCU(L) = Q(JJ)
        GO TO 402
  403  JJ = JBCU(L)
       BCU(L) = BC(JJ)
  402 IF(JBCD(L))404,405,406
  404 JJ = -JBCD(L)
      BCD(L) = EL(JJ)-EL(M+M1-1)+H(M+M1-1)
      GO TO 405
  406 JJ = JBCD(L)
      BCD(L) = BC(JJ)
  405 CALL CALC(THETA,DT,DX1(M),Q(M),A(M),B(M),I3(M),C1(M),C2(M),C3(M),
     @ H(M),EL(M),QL(M),G2,CN1(M),KE1(M),XC(M),XM(M),C(M2),R(M2),AA,
     @ LIB(L),BCU(L),BCD(L),COSP(L),BETA,IT0(L),IT1(L),IT2(L),JT,T,
     @ M1,10*(M1-1),2*M1,NS,ID(L),XMAN(M),DNDH(M))
  400 CONTINUE
C
C  THE "DO 180" LOOP ITERATES TO FIND A SET OF Qs AND As WHICH WILL
C  SATISFY THE BASIC EQUATIONS WITHIN THE TOLERANCE SPECIFIED BY THE
C  TOLERANCE PARAMETER
C
      DO 180 K=1, 50
C
C  SWEEP UPSTREAM
C
       DO 407 LL=1,NS
```

```
        L = ORDER(LL)
        M = NODE1(L)
        M1 = NNODE(L)
        M2 = 2*M-1
        CALL NEW(R(M2),Q(M),A(M),B(M),H(M),EL(M),I3(M),C1(M),C2(M),C3(M)
     @  ,C(M2),IT0(L),IT1(L),IT2(L),JT,T,M1,2*M1,NS,Z(M))
  407  CONTINUE
C
C  CHECK EACH OF THE COMPUTED DEPARTURES FOR Q AND A.  IF ANY IS LARGER
C  THAN THE COMPUTED TOLERANCE FOR EACH, GO FOR ANOTHER ITERATION,
C  A MAXIMUM OF 50.
C
       DO 160 I=1,IND1
        IF(ABS(R(2*I-1)).GT.RMSQ.OR.ABS(R(2*I)).GT.RMSA)GO TO 170
  160  CONTINUE

C  AT THIS POINT COMPUTATIONS FOR A SINGLE TIME STEP ARE COMPLETE.
C  COMPUTATIONS ARE RESTARTED AT STATEMENT 80, LOCATED WHERE THE TIME
C  MARCH BEGINS.
       GO TO 80
C
  170 CONTINUE

C   TO EXAMINE NODE AT WHICH NON-CONVERGENCE IS OCCURRING THE
C   FOLLOWING LINE HAS BEEN ADDED TO THE CODE AS A DIAGNOSTIC.

      IF(K.GE.25)THEN
        WRITE(7,*) 'Z(I)= ',Z(I),' ELAPSE= ',ELAPSE,' TOLER TST'
        WRITE(7,*) 'ABS(R(2*I-1))= ',ABS(R(2*I-1)),' RMSQ= ',RMSQ
        WRITE(7,*) 'ABS(R(2*I)= ',ABS(R(2*I)),' RMSA= ',RMSA
      END IF
C
C ***************************************************************
C THIS LOOP IS IDENTICAL TO THE "DO 165" LOOP ABOVE.  IT RECOMPUTES  *
C MANNING'S n AND CN1(I) USING NEW H(I) VALUES, WITHIN A TIME STEP.  *
C ***************************************************************
      DO 166 I=1,IND1
        IF(AX(I).NE.0.0.AND.DNDH(I).NE.0.0)THEN
            XMAN(I)=AX(I) - DNDH(I)*H(I)
          IF(XMAN(I).LE.0.01)THEN
            WRITE(7,177)I
             XMAN(I)=0.01
          END IF
         CN1(I)=XMAN(I)**2*GR/2.2
        END IF
 166  CONTINUE
C
C THE "DO 408" LOOP RESETS BOUNDARY CONDITIONS FOR CORRECTED FLOWS
C AND SOLVES THE SYSTEM OF EQUATIONS FOR NEW DEPARTURES
C
      DO 408 LL=1,NS
       L = ORDER(NS-LL+1)
       M = NODE1(L)
       M1 = NNODE(L)
       M2 = 2*M-1
```

```
      IF(JBCU(L).GE.0)GO TO 409
      JJ = -JBCU(L)
      BCU(L) = Q(JJ)
  409 IF(JBCD(L).GE.0)GO TO 410
      JJ = -JBCD(L)
      BCD(L) = EL(JJ)-EL(M+M1-1)+H(M+M1-1)
  410 CALL ITER(THETA,DT,DX1(M),Q(M),A(M),B(M),I3(M),C1(M),C2(M),C3(M),
     @ H(M),EL(M),QL(M),G2,CN1(M),KE1(M),XC(M),XM(M),C(M2),R(M2),AA,
     @ LIB(L),BCU(L),BCD(L),COSP(L),BETA,IT0(L),IT1(L),IT2(L),JT,T,
     @ M1,10*(M1-1),2*M1,NS,ID(L),XMAN(M),DNDH(M))
  408 CONTINUE
  180 CONTINUE
      WRITE(6,190)
      STOP 7
  200 CONTINUE
      RETURN
   20 FORMAT(12F6.0)
   30 FORMAT(1X,20A4)
   40 FORMAT(1X,12F10.3)
  100 FORMAT('1','DYNAMIC RIVERINE WATER QUALITY MODEL',',',
     @  'CE-QUAL-RIV1',' RIV1H ',A9)
  110 FORMAT('0',20A4)
  120 FORMAT('0','ELAPSED TIME ',I3,' HOURS ',I2,' MINUTES'.5X,'TIME ',
     @      'STEP ',I5,' SEGMENT',I3,' ',10A4)
  130 FORMAT('0',T7,'RIVER',7X,'  FLOW  ',7X,'  AREA  ',7X,'  WIDTH ',
     @       7X,'  DEPTH ',7X,'   WSEL',10X,'MANNINGS N',/,
     @       ' ',T8,'MILE',7X,'  (CFS) ',7X,' (SQ FT)',7X,' (FEET) ',
     @       7X,' (FEET) ',7X,' (FEET) ')
  140 FORMAT(' ',I2,F9.2,6F15.3)
  150 FORMAT(7F10.0)
  190 FORMAT('0THE ITERATION HAS FAILED TO CONVERGE IN 50 STEPS. RUN',
     @ ' ABORTED')
      END
C
C        SUBROUTINE CALC
C
C        PURPOSE
C           TO CALCULATE THE RESIDUALS FROM THE GOVERNING EQUATIONS
C           TO CALCULATE THE PARTIAL DERIVATIVES OF THE GOVERNING EQUATION
C           TO SET UP A VECTOR OF THE RESIDUALS
C           TO SET UP A MATRIX OF THE PARTIAL DERIVATIVES
C           TO SOLVE THIS SYSTEM OF EQUATIONS FOR THE DIFFERENCES BETWEEN
C           THE VALUES OF Q AND A IN TWO CONSECUTIVE ITERATIONS
C
C        DESCRIPTION OF PARAMETERS
C           XC     - THE J TERMS FROM THE CONTINUITY EQUATION
C           XM     - THE J TERMS FROM THE MOMENTUM EQUATION
C           R1     - THE RESIDUALS FROM THE CONTINUITY EQUATION
C           R2     - THE RESIDUALS FROM THE MOMENTUM EQUATION
C           R      - A COLLECTION OF ALL OF THE RESIDUALS IN THE PROPER
C                    ORDER FOR THE VECTOR
C           F      - THE CONTINUITY EQUATION
C           G      - THE MOMENTUM EQUATION
C           DFQ    - PARTIAL DIFFERENTIAL OF F WITH RESPECT TO Q(I)
C           DFA    - PARTIAL DIFFERENTIAL OF F WITH RESPECT TO A(I)
```

```
C           DFQ1    - PARTIAL DIFFERENTIAL OF F WITH RESPECT TO Q(I+1)
C           DFA1    - PARTIAL DIFFERENTIAL OF F WITH RESPECT TO A(I+1)
C           DGQ     - PARTIAL DIFFERENTIAL OF G WITH RESPECT TO Q(I)
C           DGA     - PARTIAL DIFFERENTIAL OF G WITH RESPECT TO A(I)
C           DGQ1    - PARTIAL DIFFERENTIAL OF G WITH RESPECT TO Q(I+1)
C           DGA1    - PARTIAL DIFFERENTIAL OF G WITH RESPECT TO A(I+1)
C           AA      - A COLLECTION OF ALL OF THE PARTIAL DERIVATIVES IN
C                     THE PROPER ORDER FOR THE BANDED MATRIX
C           DNDH    - PARTIAL OF N WITH RESPECT TO H
C  ***************************************************************
C
      SUBROUTINE CALC(THETA,DT,DX1,Q,A,B,I3,C1,C2,C3,H,EL,QL,G2,CN1,KE1,
     @XC,XM,C,R,AA,LIB,BCU,BCD,COSP,BETA,IT0,IT1,IT2,JT,T,IND1,IND2,
     @IND3,NS,ID,XMAN,DNDH)
      REAL KE1
      LOGICAL FLIP1,FLIP2
      DIMENSION Q(IND1),A(IND1),B(IND1),H(IND1),C1(IND1),C2(IND1),
     @C3(IND1),EL(IND1),QL(IND1),CN1(IND1),KE1(IND1),R(IND3),AA(IND2),
     @XC(IND1),XM(IND1),DX1(IND1),C(IND3),I3(IND1),JT(NS),T(4,NS),
     @ID(NS),XMAN(IND1),DNDH(IND1)
C
C  THE PREVIOUS TIME STEP COMPONENTS OF THE GOVERNING EQUATIONS
C
      MP=IND1-1
      Q0 = Q(1)
      A0 = A(1)
      P0 = (B(1)/A0)**(4./3.)*ABS(Q0)/A0
      D1 = (1.-THETA)*DT
      DO 10 I=1,MP
       D2 = D1/DX1(I)*2.
       Q1 = Q(I+1)
       A1 = A(I+1)
       P1 = (B(I+1)/A1)**(4./3.)*ABS(Q1)/A1
       DE = EL(I+1)-EL(I)
       XC(I)= -A0-A1+D2*(Q1-Q0)-D1*(QL(I)+QL(I+1))
       XM(I)= -Q0-Q1+D2*(BETA*(Q1*Q1/A1-Q0*Q0/A0)+G2*(A1+A0)*DE)
     @        +D1*(CN1(I)*P0*Q0 + (CN1(I+1)*P1*Q1))
       IF(KE1(I).GT.0)XM(I)=XM(I)+D2/8.*(A0+A1)*KE1(I)*(Q1*Q1/(A1*A1)
     @                      +Q0*Q0/(A0*A0))
       P0 = P1
       Q0 = Q1
       A0 = A1
   10 CONTINUE
      XC(IND1) = D1*Q1
      XM(IND1) = D1*Q1*Q1/A1*COSP
C
C  THIS IS THE ENTRY POINT FOR ALL SUCCEEDING ITERATIONS OF THE
C  NEWTON-RAPHSON PROCEDURE
C
      ENTRY ITER(THETA,DT,DX1,Q,A,B,I3,C1,C2,C3,H,EL,QL,G2,CN1,KE1,XC,
     @XM,C,R,AA,LIB,BCU,BCD,COSP,BETA,IT0,IT1,IT2,JT,T,IND1,IND2,IND3,
     @NS,ID,XMAN,DNDH)
C
C  THE UPSTREAM BOUNDARY CONDITIONS
C
```

```
      IF(LIB.GT.3)GO TO 30
      EL(1) = EL(1)-H(1)+BCU
      H(1) = BCU
      IF(I3(1))20,14,15
   14 A(1)=C1(1)*H(1)+C2(1)*H(1)**C3(1)
      B(1)=C1(1)+C2(1)*C3(1)*H(1)**(C3(1)-1.)
      GO TO  40
   15 A(1)=C1(1)*H(1)+C2(1)*H(1)**I3(1)
      B(1)=C1(1)+C2(1)*C3(1)*H(1)**(I3(1)-1)
      GO TO  40
   20 B(1)=2.*C2(1)/C1(1)*SQRT((2.*C1(1)-H(1))*H(1))
      A(1)=C1(1)*C2(1)*ACOS(1.-H(1)/C1(1))-B(1)*(C1(1)-H(1))/2.
      GO TO  40
   30 Q(1)=BCU
   40 CONTINUE
C
C  DOWNSTREAM BOUNDARY CONDITIONS
C
      GO TO(43,60,70,43,60,70),LIB
   43 EL(IND1) = EL(IND1)-H(IND1)+BCD
      H(IND1)=BCD
      AA(IND2) = 0.
      R(IND3) = 0.
      IF(I3(IND1))50,44,45
   44 A(IND1)=C1(IND1)*H(IND1)+C2(IND1)*H(IND1)**C3(IND1)
      B(IND1)=C1(IND1)+C2(IND1)*C3(IND1)*H(IND1)**(C3(IND1)-1.)
      GO TO  51
   45 A(IND1)=C1(IND1)*H(IND1)+C2(IND1)*H(IND1)**I3(IND1)
      B(IND1)=C1(IND1)+C2(IND1)*C3(IND1)*H(IND1)**(I3(IND1)-1)
      GO TO  51
   50 B(IND1)=2.*C2(IND1)/C1(IND1)*H(IND1)*SQRT(2.*C1(IND1)/H(IND1)-1.)
      A(IND1)=C1(IND1)*C2(IND1)*ACOS(1.-H(IND1)/C1(IND1))
     @        -B(IND1)*(C1(IND1)-H(IND1))/2.
   51 C(IND3) = -B(IND1)
      GO TO  80
   60 Q(IND1)=BCD
      R(IND3)=0.
      AA(IND2) = 0.
      GO TO  80
   70 EXPO = BCD
      COEF = COSP
      R(IND3)=-(COEF*Q(IND1)**EXPO-H(IND1))
      DBCQ=COEF*EXPO*Q(IND1)**(EXPO-1.)
      DBCA=-1./B(IND1)
      AA(IND2-1)=DBCQ
      AA(IND2)=DBCA
   80 CONTINUE
      IF(I3(1))90,84,85
   84 DBDH=C2(1)*C3(1)*(C3(1)-1.)*H(1)**(C3(1)-2.)
      DNDH2 = DNDH(1)
      GO TO 100
   85 DBDH=C2(1)*C3(1)*(C3(1)-1.)*H(1)**(I3(1)-2)
      DNDH2 = DNDH(1)
      GO TO 100
   90 DBDH=4.*(C2(1)/C1(1))**2*(C1(1)-H(1))/B(1)
```

```
      DNDH2 = DNDH(1)
  100 CONTINUE
C
C  THE GOVERNING EQUATIONS
C
      MP = IND1-1
      Q0 = Q(1)
      A0 = A(1)
      P0 = (B(1)/A0)**(4./3.)*ABS(Q0)/A0
      IR = 2
      IA = 1
      D1 = THETA*DT
      DO 130 I=1,MP
       D2 = D1/DX1(I)*2.
       Q1 = Q(I+1)
       A1 = A(I+1)
       P1 = (B(I+1)/A1)**(4./3.)*ABS(Q1)/A1
       CN = CN1(I)*D1
       CN2 = CN1(I+1)*D1
       DE = EL(I+1)-EL(I)
       R(IR)=-(A0+A1+D2*(Q1-Q0)-D1*(QL(I)+QL(I+1))+XC(I))
       R(IR+1)=-(Q0+Q1+D2*(BETA*(Q1*Q1/A1-Q0*Q0/A0)+G2*(A0+A1)*DE)
     @           +(CN*Q0*P0+CN2*Q1*P1)+XM(I))
       DFQ = -D2
       DFA = 1.
       DFQ1 = D2
       DFA1 = 1.
       DGQ = 1.-2.*D2*BETA*Q0/A0 + CN*P0
       DGA = D2*(BETA*Q0*Q0/(A0*A0)+G2*(DE-(A1+A0)/B(I)))
     @ +CN*Q0*P0/3.*(-7./A0+4.*DBDH/(B(I)*B(I)+6./XMAN(I)*B(I)*DNDH2))
       DGQ1 = 1.+2.*D2*BETA*Q1/A1+ CN2*P1
       IF(I3(I+1))110,104,105
  104  DBDH=C2(I+1)*C3(I+1)*(C3(I+1)-1.)*H(I+1)**(C3(I+1)-2.)
       DNDH2=DNDH(I+1)
       GO TO 120
  105  DBDH=C2(I+1)*C3(I+1)*(C3(I+1)-1.)*H(I+1)**(I3(I+1)-2)
       DNDH2=DNDH(I+1)
       GO TO 120
  110  DBDH=4.*(C2(I+1)/C1(I+1))**2*(C1(I+1)-H(I+1))/B(I+1)
       DNDH2=DNDH(I+1)
  120  CONTINUE
       DGA1 = D2*(-BETA*Q1*Q1/(A1*A1)+G2*(DE+(A0+A1)/B(I+1)))
     @ +CN2*Q1*P1/3.*(-7./A1+4.*DBDH/(B(I+1)*B(I+1)+6./(XMAN(I+1)
     @ *B(I+1))*DNDH2))
        IF(DGA1.EQ.0.0) DGA1 = 1.0E-5
       IF(KE1(I).LE.0.)GO TO 125
       R(IR+1)=R(IR+1)-D2/8.*(A0+A1)*(Q1*Q1/(A1*A1)
     @          +Q0*Q0/(A0*A0))*KE1(I)
       DGQ=DGQ+KE1(I)*D2/4.*(A0+A1)*Q0/(A0*A0)
       DGA=DGA+D2/8.*KE1(I)*(Q1*Q1/(A1*A1)-Q0*Q0/(A0*A0)
     @     *(1.+2.*A1/A0))
       DGQ1=DGQ1+KE1(I)*D2/4.*(A0+A1)*Q1/(A1*A1)
       DGA1=DGA1+D2/8.*KE1(I)*(Q0*Q0/(A0*A0)-Q1*Q1/(A1*A1)
     @     *(1.+2.*A0/A1))
  125  AA(IA) = DFQ
```

```
      AA(IA+1) = DFA
      AA(IA+2) = DFQ1
      AA(IA+3) = DFA1
      AA(IA+4) = DGQ
      AA(IA+5) = DGA
      AA(IA+6) = DGQ1
      AA(IA+7) = DGA1
      IR = IR+2
      IA = IA+10
      Q0 = Q1
      A0 = A1
      P0 = P1
  130 CONTINUE
C
C  ACCOUNT FOR TRIBS AND FLIP COLUMNS IF NECESSARY
C
      IF(IT1.EQ.0)GO TO 71
      DO 72 L=IT1,IT2
      I = JT(L)
      DX = DX1(I-1)
      IR = 2*(I-1)
      IA = 5*IR
      B0 = B(I)
      AA(IA-6) = AA(IA-6)+T(1,L)/(DX*B0)
      AA(IA-2) = AA(IA-2)+T(3,L)/(DX*B0)
      R(IR) = R(IR)+T(2,L)/DX
      R(IR+1) = R(IR+1)+T(4,L)/DX
      DX = DX1(I)
      AA(IA+2) = AA(IA+2)+T(1,L)/(DX*B0)
      AA(IA+6) = AA(IA+6)+T(3,L)/(DX*B0)
      R(IR+2) = R(IR+2)+T(2,L)/DX
      R(IR+3) = R(IR+3)+T(4,L)/DX
   72 CONTINUE
   71 FLIP1 = LIB.LE.3
      FLIP2 = LIB.EQ.2.OR.LIB.EQ.5
      IF(.NOT.FLIP1)GO TO 73
      SAVE = AA(1)
      AA(1) = AA(2)
      AA(2) = SAVE
      SAVE = AA(5)
      AA(5) = AA(6)
      AA(6) = SAVE
   73 IF(.NOT.FLIP2) GO TO 74
      SAVE = AA(IND2-2)
      AA(IND2-2) = AA(IND2-3)
      AA(IND2-3) = SAVE
      SAVE = AA(IND2-6)
      AA(IND2-6) = AA(IND2-7)
      AA(IND2-7) = SAVE
   74 CONTINUE
      CALL MAT5(AA,C(2),R(2),IND2,IND3-1)
C
C  UNDO FLIPPING AND PLACE DATA INTO THE T-ARRAY
C
      IF(.NOT.FLIP1)GO TO 75
```

```
      R(1) = R(2)
      R(2) = 0.
   75 IF(.NOT.FLIP2)GO TO 76
      R(IND3) = R(IND3-1)
      R(IND3-1) = 0.
      GO TO 77
   76 IF(IT0.EQ.0)GO TO 77
      T(1,IT0) = D1*C(IND3-1)
      T(2,IT0) = D1*(R(IND3-1)+Q1)+XC(IND1)
      T(3,IT0) = D1*COSP*Q1/A1*(2.*C(IND3-1)+Q1*B(IND1)/A1)
      T(4,IT0) = D1*COSP*Q1/A1*(2.*R(IND3-1)+Q1)+XM(IND1)
   77 CONTINUE
      RETURN
      END
      SUBROUTINE MAT5(AA,C,R,M0,M)
      DIMENSION AA(M0),C(M),R(M)
      M1=M-1
C
C  ELIMINATE EXTREME DIAGONALS
C
      J=1
      DO 1 I=1,M1,2
       D = -AA(J+3)/AA(J+7)
       AA(J) = AA(J)+AA(J+4)*D
       AA(J+1) = AA(J+1)+AA(J+5)*D
       AA(J+2) = AA(J+2)+AA(J+6)*D
       R(I) = R(I) + R(I+1)*D
       D = -AA(J+4)/AA(J)
       AA(J+5) = AA(J+5) + AA(J+1)*D
       AA(J+6) = AA(J+6) + AA(J+2)*D
       R(I+1) = R(I+1) + R(I)*D
       J =J+10
    1 CONTINUE
C
C  ELIMINATE LOWER CODIAGONAL
C
       J = 6
      DO 2 I=2,M1
       D = -AA(J)/AA(J-4)
       AA(J+1) = AA(J+1) + AA(J-3)*D
       R(I) = R(I) + R(I-1)*D
       J = J+5
    2 CONTINUE
C
C  IN THE CASE OF A NON-TRIBUTARY WITH A RATING CURVE DOWNSTREAM,
C  THE BOTTOM ROW OF THE MATRIX NEEDS WORK
C
      IF(AA(M0).EQ.0)GO TO 5
      D = -AA(M0-1)/AA(M0-3)
      R(M) = (R(M)+R(M1)*D)/(AA(M0)+AA(M0-2)*D)
C
C  ELIMINATE UPPER CODIAGONAL AND NORMALIZE MATRIX
C
    5 J = 5*M1 - 2
       DO 3 II=1,M1
```

```
      I = M1-II+1
      C(I) = C(I+1)* (-AA(J)/AA(J-1))
      R(I) = (R(I)-AA(J)*R(I+1))/AA(J-1)
      J = J-5
    3 CONTINUE
      RETURN
      END
      SUBROUTINE NEW(R,Q,A,B,H,EL,I3,C1,C2,C3,C,IT0,IT1,IT2,JT,T,M1,M2,
     @NS,Z)
      DIMENSION R(M2),A(M1),Q(M1),B(M1),C1(M1),C2(M1),
     @C3(M1),H(M1),EL(M1),I3(M1),C(M2),JT(NS),T(4,NS),Z(M1)
      IF(IT0.NE.0)D = T(1,IT0)
      DO 60 I=1,M1
      II = 2*I
      IF(IT0.EQ.0)GO TO 1
      R(II-1) = R(II-1)-C(II-1)*D
      R(II) = R(II)-C(II)*D
    1 Q(I)=Q(I)+R(II-1)
      A(I)=A(I)+R(II)
      H1 = H(I)
C
C     DIAGNOSTIC CHECK TO SEE IF DEPTH GOES TO ZERO
C
      DO 30 K=1,10
      H0=H1
      IF(H0.LE.0)WRITE(7,819)
  819 FORMAT(2X,'ALERT - COMPUTED DEPTH LESS THAN ZERO')
      IF(H0.LE.0) WRITE(7,818) Z(I),H0,K
  818 FORMAT(10X,'NODE ELEV.=', F6.2,5X, 'H0=',F5.2, 5X, 'K=', I4)
      IF(I3(I))10,4,5
    4 B0=C1(I)+C2(I)*C3(I)*H0**(C3(I)-1.)
      H1=H0-(C1(I)*H0+C2(I)*H0**C3(I)-A(I))/B0
      GO TO 20
    5 B0=C1(I)+C2(I)*C3(I)*H0**(I3(I)-1)
      H1=H0-(C1(I)*H0+C2(I)*H0**I3(I)-A(I))/B0
      GO TO 20
   10 B0=2.*C2(I)/C1(I)*SQRT((2.*C1(I)-H0)*H0)
      H1=H0-(C1(I)*C2(I)*ACOS(1.-H0/C1(I))-B0*(C1(I)-H0)/2.
     @          -A(I))/B0
   20 IF(ABS((H1-H0)/H0).LE.0.01) GO TO 50
   30 CONTINUE
      WRITE(6,40)
      STOP 7
   50 B(I)=B0
      EL(I) = H1+Z(I)
      H(I) = H1
   60 CONTINUE
C
C  PASS DATA TO TRIBS
C
      IF(IT1.EQ.0)GO TO 70
      DO 80 L=IT1,IT2
      K = JT(L)
      T(1,L) = R(2*K)/B(K)
   80 CONTINUE
```

```
70 RETURN
40 FORMAT('0THE STAGE-AREA EQUATION HAS FAILED TO CONVERGE IN',
  @        ' 10 ITERATIONS.  RUN ABORTED.')
   END
   SUBROUTINE BUBBLE(IT0,IT1,IT2,JT,NS)
   DIMENSION IT0(NS),IT1(NS),IT2(NS),JT(NS)
   DO 1 L=1,NS
    IF(IT2(L).LE.IT1(L))GO TO 1
    LAST = IT2(L)
 2  LIMIT = LAST-1
    LAST = IT1(L)
    M1 = LAST
    DO 3 I=M1,LIMIT
     IF(JT(I).LE.JT(I+1))GO TO 3
     ITEMP = JT(I)
     JT(I) = JT(I+1)
     JT(I+1) = ITEMP
     LAST = I
     DO 5 M=1,NS
      IF(IT0(M).NE.I)GO TO 6
      IT0(M) = I+1
      GO TO 5
 6    IF(IT0(M).EQ.I+1)IT0(M) = I
 5   CONTINUE
 3 CONTINUE
   IF(LAST.GT.M1)GO TO 2
 1 CONTINUE
   RETURN
   END
```

APPENDIX C: LISTING OF THE WATER QUALITY CODE RIV1Q

```
C   ************************************************************************
C   *                                                                      *
C   *                   CE-QUAL-RIV1       VAX VERSION                   *
C   *                  RIV1Q CODE - WATER QUALITY MODEL                    *
C   *                                                                      *
C   *         THIS VERSION OF CE-QUAL-RIV1Q CORRESPONDS TO THE FIRST       *
C   *         RELEASE VERSION OF THE USER MANUAL                           *
C   *         THIS MODEL WAS ORIGINALLY DEVELOPED BY OHIO STATE U.,        *
C   *            BEDFORD, SYKES, AND LIBICKI.                              *
C   *         THIS VERSION HAS BEEN MODIFIED BY:                           *
C   *            M. DORTCH, T. SCHNEIDER, M. ZIMMERMAN,                    *
C   *            AND J. MARTIN OF THE                                      *
C   *            WATER QUALITY MODELING GROUP                              *
C   *            USAE WATERWAYS EXPERIMENT STATION                         *
C   *            VICKSBURG, MISSISSIPPI, AND                               *
C   *            D. GRIFFIN, JR.                                           *
C   *            LOUISIANA TECH UNIVERSITY                                 *
C   *            RUSTON, LOUISIANA                                         *
C   ************************************************************************

      PROGRAM RIV1Q

      COMMON DJ(500000)
      COMMON/ABLOCK/ IND1,  IND2,  IND3,  IND4,  IND5,  NS

C     FILE 5 IS THE INPUT DATA FILE

      OPEN (UNIT=5, FILE= 'FILE5', STATUS='OLD')

C     FILE 6 IS THE OUTPUT DATA FILE

      OPEN (UNIT=6, FILE= 'FILE6', STATUS='NEW')

C     FILE2 CONTAINS METEOROLOGICAL DATA

      OPEN (UNIT=2, FILE= 'FILE2', STATUS='OLD',READONLY)

C     BINARY FILE 8 CONTAINS HYDRODYNAMIC DATA FROM RIV1H

      READ (8) MNODE,  MTIME,  NS

      IND1 = MNODE
      IND2 = MTIME-1
      IND3 = IND2*NS
      IND4 = 4*IND1
      IND5 = 5*IND1

      N1  = 1   + NS
      N2  = N1  + NS
      N3  = N2  + NS
      N4  = N3  + NS
      N5  = N4  + NS
      N6  = N5  + NS
      N7  = N6  + NS
      N8  = N7  + NS
```

```
      N9  = N8  + NS
      N10 = N9  + NS
      N11 = N10 + NS
      N12 = N11 + NS
      N13 = N12 + NS
      N14 = N13 + 42*NS
      N15 = N14 + 10*NS
      N16 = N15 + 31*NS
      N17 = N16 + 10*IND1
      N18 = N17 + 10*IND1
      N19 = N18 + 10*IND1
      N20 = N19 + 10*IND1
      N21 = N20 + 10*IND1
      N22 = N21 + 10*NS
      N23 = N22 + IND1
      N24 = N23 + IND1
      N25 = N24 + IND1
      N26 = N25 + IND1
      N27 = N26 + IND1
      N28 = N27 + IND1
      N29 = N28 + IND1
      N30 = N29 + 4*IND1
      N31 = N30 + 10*IND3
      N32 = N31 + 6*IND1
C
C              C A L L   S U B R O U T I N E   'M A I N 2'
C
      CALL MAIN2(DJ(1),DJ(N1), DJ(N2), DJ(N3), DJ(N4), DJ(N5), DJ(N6),
     * DJ(N7), DJ(N8), DJ(N9), DJ(N10),DJ(N11),DJ(N12),DJ(N13),DJ(N14),
     * DJ(N15),DJ(N16),DJ(N17),DJ(N18),DJ(N19),DJ(N20),DJ(N21),DJ(N22),
     * DJ(N23),DJ(N24),DJ(N25),DJ(N26),DJ(N27),DJ(N28),DJ(N30),DJ(N25),
     * DJ(N29),DJ(N31),DJ(N32))

      STOP
      END


C
C     ..........................................
C     ..    S U B R O U T I N E   M A I N 2     ..
C     ..........................................

      SUBROUTINE MAIN2(ID,ID0,ORDER,IDAM,DAMK,NNODE,NODE1,IBC,
     *JBC,IT0,IT1,IT2,JT,T,SNAME,PARM,C,DC,K,SINK,INIT,CL,DX1,RMILE,QL,
     *Q,A,B,EL,BOUND,HYDRO1,HYDRO2,F,DISP)


      IMPLICIT REAL(K)

      COMMON/ABLOCK/ IND1,          IND2,          IND3,
     *               IND4,          IND5,          NS

      COMMON/STAR/   START,         TEMP
      COMMON/TIME/   STB,           STE,           DELTAT,
     *               TOFDAY,        JDYO,          LII,
```

```
     *               LMN,              CONS7,            I,
     *               J
      COMMON/PERIOD/ LMREAD,           HSTART,           ITEM

      COMMON/SUB/    JDY0,             DUC,              LSMC,
     *               LATUDC,           LONTUC,
     *               TEM0,             WIND0,            CLOUD0,
     *               ATM0,             DEPTH0,           WSE0,
     *               RHO0,             WET0,             DRY0,
     *               HFLUX,            DTEM,             FWST,
     *               HNEFSW,           OXIDAT,
     *               KFEDK,            KMNDK,            KCOLIDK

      DIMENSION    DX1(IND1),    Q(IND5),      A(IND5),B(IND5),QL(IND1),
     *EL(IND5),C(10,IND1),    DC(10,IND1),  K(10,IND1),  SINK(10,IND1),
     *INIT(10,IND1),BOUND(10,IND3),HYDRO1(IND4),HYDRO2(IND4),
     *RMILE(IND1),ID(NS), ID0(NS),ORDER(NS),IDAM(NS),DAMK(NS),NNODE(NS),
     *NODE1(NS),   IBC(NS),     JBC(NS),IT0(NS),IT1(NS),IT2(NS),JT(NS),
     *T(42,NS),    SNAME(10,NS),PARM(31,NS),  F(IND1,6),   DISP(IND1),
     *CL(10,NS),    CP(10),       TITLE(20),    SNAME0(10),   MC(10),
     *NUM(10), NUPD(200,10),DUMMY(10),QWIND(20)
C
C ##########################################################################
C # CURRENTLY LIMITED TO 200 DIFFERENT UPDATES BY NUPD                     #
C # AND 10 TRIBUTARIES BY NUM  AND NUPD, 10 VARIABLES BY DUMMY             #
C ##########################################################################
C
      CHARACTER      ITEM0*8,          ITEM1*8,          UNIT0*20,
     *               UNIT1*20
      REAL*8  BLABEL(10),BUNIT(10)/10*' (MG/L) '/,ALABEL(10)/'  TEMP  ',
     *         ' CBODNS ',' ORGAN ',' NH3N  ',' NO3N  ',' PO4   ',
     *         '   DO   ','   MN   ','   FE   ','COLIFORM'/,
     *         AUNIT(2)/'(DEG. C)',' COL/DL '/,CLABEL,CUNIT,
     *         FMT1/',  F12.1'/,FMT2/',  F12.2'/,
     *         FMT(13)/'(1X,I2,F','8.2     ',10*'        ',')       '/

      CHARACTER*9 ADATE

      INTEGER ORDER
      REAL           LAMBDA,           INIT,
     *               NH3NL,            NO3NL,            MNL
      REAL           LATUDC,           LONTUC,           LSMC

      LOGICAL QWIND,QWINDO
      LOGICAL DARK,CP/10*.FALSE./

C
C     SET DEFAULT VALUES FOR THE CONSTANTS AND SET UP "NAMELIST" INPUT
C
      DATA TEMP,BK,AK1,KOCB1,AG,E1,E2,TSW,AKN,KON,AKNX/
     *     23.0,-0.49,0.15,0.5,12.81,0.5,1.5,0.0,0.3,0.5,0.0/

      DATA KCOLIDK, KMNDK, KFEDK, OXIDAT/
     *     1.4, 0.5, 0.5, 1.0/
```

```
      DATA APO4,ADN,KNCBDN,KOCBDN,ALGAEB,KALGRO,KALGDK/
     *     0.0,0.1,0.1,0.5,0.0,0.002,0.1/

      DATA KOALDK,DAWN,SUNSET,KEXT,START,
     *     0.5,6.30,20.30,.5,0.00/,PI/3.1415o/

      DATA CSINK,ATS,ATB,TSINK,IPRINT,IPLOT,ITEM,HNEFSW/
     *     0.,0.,0.,10.,1,0,0,0./

      DATA CBODL,NH3NL,ORGANL,TEMPL,PO4L,NO3NL,DOL/
     *     0.,0.,0.,0.,0.,0.,0./

      DATA MNL, FEL, COLIL/
     *     0., 0., 0./

C
C     NEW NAMES ARE APPENDED TO THE END OF NAMELIST
C
      NAMELIST/CONST/ADN,AG,AKN,AKNX,AK1,ALGAEB ATB,ATS,APO4,BK,
     *CBODL,CLABEL,CSINK,CUNIT,DAWN,DOL,E1,E2,IPRINT,IPLOT,
     *KALGDK,KALGRO,KEXT,KNCBDN,KOALDK,KOCBDN,KOCB1,KON,NH3NL,
     *NO3NL,ORGANL,PO4L,START,SUNSET,TEMP,TEMPL,TSINK,TSIV,KOLIDK,
     *KMNDK,KFEDK,OXIDAT,MNL,FEL,COLIL.ITEM,HNEFSW,QWINDO

C     INITIALIZE QWIND TO FALSE

      DO I=1,NS
      QWIND(I)=.FALSE.
      END DO

      READ(5,10) TITLE

      READ(8) JT, IT0, IT1, IT2, ORDER, NNODE, NODE1, ID, JBC

      DO 2 LL = 1,NS
      IBC(LL) = 0
C
C     READ STREAM CARD
C
      READ(5,3) IDLL, SNAME0, IDAM0, DAMK0
    3 FORMAT(I2,10A4,I2,F10.0)
      DO 100 L0 = 1,NS
      L = L0
      IF (ID(L0).EQ.IDLL) GO TO 101
  100 CONTINUE

      WRITE(6,102) IDLL
  102 FORMAT(' ID#',I3,' NOT FOUND IN OUTPUT FROM RIV1H.')
C
      RETURN
C
  101 ID0(LL) = L
      IDAM(L) = IDAM0
      IF (IDAM0.GT.0) DAMK0 = DAMK0/0.3048
```

```
      DAMK(L) = DAMK0

      DO 103 JJJ = 1,10
 103  SNAME(JJJ,L) = SNAME0(JJJ)
C
C     READ CONSTANT CARD (NAMELIST VARIABLES)
C
      QWINDO=.FALSE.
      READ(5,CONST)
      IF(QWINDO) QWIND(LL)=.TRUE.
      IF (LL.EQ.1) GO TO 4
      M1 = L
      M2 = L
      GO TO 5
    4 M1 = 1
      M2 = NS
C
    5 CONTINUE
C
C
      DO 6 M = M1,M2
C
C     PACK SEGMENT-SPECIFIC PARAMETERS INTO THE PARM ARRAY
C     AND PERFORM CONVERSIONS
C
C TO CONVERT FROM CFS**-BK TO (CU M/DAY)**-BK
C     CONVRT     = 4.08732E-04**BK
      CONVRT     = 1.0
      PARM(1 ,M) = ADN*CONVRT
      PARM(2 ,M) = AG*3.79726E-05**E1*0.3048**E2
      PARM(3 ,M) = AK1*CONVRT
      PARM(5 ,M) = AKN*CONVRT
      PARM(6 ,M) = AKNX*CONVRT
      PARM(7 ,M) = ALGAEB*KALGRO
      PARM(8 ,M) = ALGAEB*KALGDK
      PARM(9 ,M) = APO4*CONVRT
      PARM(10,M) = ATB*CONVRT
      PARM(11,M) = ATS*CONVRT
      PARM(12,M) = BK
      PARM(13,M) = CSINK
      PARM(14,M) = E1
      PARM(15,M) = E2
      PARM(16,M) = KEXT/0.3048
      PARM(17,M) = KNCBDN
      PARM(18,M) = KOALDK
      PARM(19,M) = KOCBDN
      PARM(20,M) = KOCB1
      PARM(21,M) = KON
      PARM(22,M) = TEMP
      PARM(23,M) = TSINK
      PARM(24,M) = TSIV/0.3048
      PARM(29,M) = KCOLIDK
      PARM(30,M) = KFEDK
      PARM(31,M) = KMNDK
C
```

```
C     ASSIGN LATERAL INFLOW CONCENTRATIONS AND TEST FOR PRESENCE
C
      CL(1,M) = TEMPL
      CL(2,M) = CBODL
      CL(3,M) = ORGANL
      CL(4,M) = NH3NL
      CL(5,M) = NO3NL
      CL(6,M) = PO4L
      CL(7,M) = DOL
      CL(8,M) = MNL
      CL(9,M) = FEL
      CL(10,M) = COLIL

      DO 50 N = 1,10
      IF (CL(N,M).GT.0)CP(N) = .TRUE.
   50 CONTINUE
C

    6 CONTINUE

C     READ INITIAL CONDITIONS CARDS
C
      M1 = NODE1(L)
      M2 = M1 + NNODE(L)-1

      DO 9 I = M1, M2
      READ(5,7) (INIT(N,I),N=1,10)
    7 FORMAT(10F8.0)
C
C     READ IN DISPERSION COEFFICIENT IN FT2/SEC FOR EACH REACH
C
      READ(5,11) DISP(I)
      DISP(I) = DISP(I)*8026.8
   11 FORMAT(F10.0)
    9 CONTINUE
C
    2 CONTINUE
C
C     READ BOUNDARY CONDITIONS ID CARD
C
      READ(5,*) IBC

      NBC = 0

      DO 38 L = 1,NS
      IF(IBC(L).EQ.0)GO TO 39
      NBC = NBC + 1
   38 CONTINUE
   39 CONTINUE
C
C     READ BOUNDARY CONDITIONS CARDS AND TEST FOR PRESENCE
C
      M2 = NBC * (IND2 + 1)
C  BOUNDARY CONDITION INPUTS
        DO  201 J=1,NBC
```

```
            READ(5,81)NUM(J)
            READ(5,81)(NUPD(I,J),I=1,NUM(J))
               READ(5,7)(DUMMY(I),I=1,10)
               ICOUNT=1
               N=1
            DO 201 KK=J,M2,NBC
                  IF(ICOUNT.GT.NUPD(N,J))THEN
                     N=N+1
                     READ(5,7)(DUMMY(I),I=1,10)
                  END IF
                  ICOUNT=ICOUNT+1
              DO 201 KKK=1,10
                     BOUND(KKK,KK)=DUMMY(KKK)
  201            CONTINUE
  81           FORMAT(10I8)


      DO 8 M = 1,M2
      DO 8 N = 1,10
      IF (BOUND(N,M).GT.0)CP(N) = .TRUE.
    8 CONTINUE
C
C  UNSCRAMBLE REFERENCES
C
      DO 31 L = 1,NS
      DO 31 M = 1,NS
      IF (IBC(L).EQ.ID(M)) IBC(L) = M
   31 CONTINUE
C
C  CONSTRUCT CROSS-REFERENCE BOUNDARY CONDITIONS DIRECTORY
C
      DO 32 L = 1,NBC
      IBCL    = IBC(L)
      IF (JBC(IBCL).GE.0) GO TO 33
      WRITE (6,34) ID(IBCL)
   34 FORMAT(' SEGMENT #',I3,' MAY NOT RECEIVE INPUT BOUNDARY',
     *       ' CONDITIONS.')
C
C
      RETURN

   33 JBC(IBCL) = L
   32 CONTINUE
C
C   READ IN DISTANCE INCREMENTS AND LATERAL INFLOWS FROM THE
C   HYDRODYNAMIC MODEL
C
      READ (8)  DX1, QL, RMILE
C  CONVERT ENGLISH TO METRIC UNITS
      DO 47 I = 1,IND1
      DX1(I)  = DX1(I)*0.3048
      QL(I)   = QL(I)*8026.85
   47 CONTINUE
C
C   CALL IN THE CURRENT DATE FROM THE CPU
```

```
C
      CALL DATE(ADATE)
C
C   CONVERT TIMES FROM HH.MM TO DAYS
C
      START  = START/14.4-AINT(START)/36.
      SUNSET = SUNSET/14.4-AINT(SUNSET)/36.
      DAWN   = DAWN/14.4-AINT(DAWN)/36.
      LAMBDA = SUNSET-DAWN
      ELAPSE = 0.
      CLOCK  = START
C
      HSTART = START * 24.
C
C  WRITE OUT INPUT CONCENTRATIONS
C
      MBOUND = IND2*NBC
      WRITE(6,10)TITLE
      WRITE(6,135) ALABEL
      WRITE(6,140) (I,(INIT(N,I),N=1,10),I=1,IND1)

C     WRITE OUT BOUNDARY CONDITIONS
         DO 301 I=1,NBC
      WRITE(6,142)IBC(I)
      WRITE(6,81)(NUPD(J,I),J=1,NUM(I))
      WRITE(6,136) ALABEL
      WRITE(6,141)((BOUND(J,(NUPD(KK,I)*NBC-NBC+I)),J=1,10),KK=1,NUM(I))
301      CONTINUE
C
C   THERE IS AN UPPER LIMIT OF ONE BASE RUN AND FOUR SENSITIVITY
C   ANALYSIS RUNS
C
C              S E N S I T I V I T Y   C O M P U T A T I O N
C
      DO 520 ICYCLE = 1,5
      WRITE(6,CONST)
C
C   ASSIGN INITIAL CONDITIONS (N,I)=INIT(N,I) AND TEST FOR PRESENCE
C
      DO 51 N = 1,10
      DO 51 I = 1,IND1
      C(N,I)  = INIT(N,I)
      IF (C(N,I).GT.0.0) CP(N) = .TRUE.
   51 CONTINUE

C   IN SITU SOURCES

      IF (ALGAEB.GT.0.0) CP(4) = .TRUE.
      IF (ALGAEB.GT.0.0) CP(6) = .TRUE.
      IF (ALGAEB.GT.0.0) CP(7) = .TRUE.
C
C
      MM = 0
      DO 630 N = 1,10
      IF(.NOT.CP(N)) GO TO 630
```

```
      MM = MM + 1
      MC(MM) = N
      IF (N.NE.2.AND.N.NE.7) FMT(MM + 2) = FMT2
      IF (N.EQ.7.OR.N.EQ.2)  FMT(MM + 2) = FMT1
      BLABEL(MM) = ALABEL(N)
      IF (N .EQ. 1) BUNIT(MM) = AUNIT(1)
C
C
      IF (N .EQ. 10) BUNIT(MM) = AUNIT(2)
C
C
  630 CONTINUE

      MAX = MM
C
C   CONSTRUCT A CUBIC SPLINE THROUGH THE INITIAL CONCENTRATIONS
C   FOR EACH SEGMENT
C
C                  C A L L     S U B R O U T I N E     'SPLINE'
C
      DO 55 L = 1,NS
      M1 = NODE1(L)
      M2 = NNODE(L)
      CALL SPLINE(C(1,M1),DC(1,M1),CP,DX1(M1),F,M2)
   55 CONTINUE
C
C   BEGIN TIME MARCH
C
C                    T I M E   S T E P   C O M P U T A T I O N
C
      MTIME = IND2 + 1
 3001 FORMAT(3X,'MTIME=',I4)
C
      DO 499 J = 1,MTIME
C

      MBC = (J-1)*NBC
C
C   READ HYDRODYNAMICS, ALTERNATING BETWEEN HYDRO1 AND HYDRO2
C
      IF (MOD(J,2)) 110,110,111
  110 READ(8)DT,HYDRO1

      L0 = IND1*4
      L1 = 0
      GO TO 112
  111 READ(8)DT,HYDRO2

      L0 = 0
      L1 = IND1*4
  112 CONTINUE
C
      DT = DT/86400.
      M1 = L1 + 1
      M2 = L1 + IND1
```

```
      DO 113 I = M1,
      Q(I) = Q(I)*24.
      A(I) = A(I)*0.0929034
      B(I) = B(I)*0.3048
      EL(I) = EL(I)*0.3048
  113 CONTINUE
C
C   SET  RUNNING PARAMETERS
C
      IF (J.EQ.1) DT = 0.
      ELAPSE = ELAPSE + DT
      CLOCK  = AMOD(CLOCK + DT,1.)
      DARK   = CLOCK.GE.SUNSET.OR.CLOCK.LE.DAWN
      IF (DARK) SINI = 0.
      IF (.NOT.DARK) SINI=SIN(PI*(CLOCK-DAWN)/LAMBDA)
C
      DELTAT = DT * 24.

      IF (J .EQ. 1) STB = HSTART
      STE = STB + DELTAT
C
      IF (J .EQ. 1) THEN
      TOFDAY = HSTART
      ELSE
      TOFDAY = TOFDAY  + DELTAT
      END IF

C   "DO 505" LOOP CALLS SUBROUTINE SEG AND PROCESSES SEGMENTS IN A
C    DOWNSTREAM DIRECTION. THIS LOOP IS EXECUTED DURING EACH TIME STEP.

C
      DO 505 LL = 1,NS
C
      LMN = LL

      L   = ORDER(NS-LL + 1)
      M   = NODE1(L)
      M0  = L0 + M
      M1  = L1 + M
C
C  CALL SEG WITH BOUNDARY CONDITIONS TAKEN FROM BOUND, IF APPLICABLE

C  JBC(L) = 0 MEANS THAT A NCECESSARY BOUNDARY CONDITION WAS NOT FOUND.
C  THE PROGRAM GOES TO STATEMENT 601 AND ABORTS.

      IF (JBC(L)) 600,601,602

  602 JJ = JBC(L) + MBC

C
C            C A L L    S U B R O U T I N E    'SEG'
C ...NOTE:  IN THE CALLS BELOW PARM(1,L) IS SENT TO SEG.  HOWEVER PARM
C ........  IS RECEIVED AS A 1-D ARRAY DIMENSIONED TO 31 LOCATIONS
C ........  THIS CAUSES PARM(1,L) TO ACTUALLY SEND THE FIRST 31 VALUES
```

```
C ........  CORRESPONDING TO THE LEFT SUBSCRIPT FOR EACH VALUE OF L.
C
      CALL SEG(DX1(M),Q(M0),A(M0),B(M0),EL(M0),Q(M1),A(M1),B(M1),
     *EL(M1),QL(M),
     *C(1,M),DC(1,M),K(1,M),SINK(1,M),BOUND(1,JJ),CP,CL(1,L),
     *PARM(1,L),SINI,NNODE(L),IT0(L),IT1(L),IT2(L),JT,T,DT,
     *DISP(M),F,NS,QWIND(L))
      GO TO 505
C
  601 WRITE(6,603) ID(L)
  603 FORMAT(' MISSING BOUNDARY CONDITION FOR SEGMENT',I3,
     *  '.  RUN ABORTED.')

C
      RETURN
C
C   PASS BOUNDARY CONDITIONS THROUGH CONTROL STRUCTURES IN OTHER CASES
C
  600 JJ = -JBC(L)

C
C   REAERATION OVER DAMS.  THIS SECTION HAS BEEN MODIFIED
C   GREATLY TO ACCOMMODATE EDM REAERATION.

      IF (CP(1)) TEMP = C(1,M)
      IF (.NOT.CP(1)) TEMP = PARM(22,L)
      DOSAVE = C(7,JJ)
      IF (IDAM(L)-1) 507,508,508


C       EDM REAERATION BELOW

  508   CONTINUE
        CTR = DAMK(L) * 1.022 ** (TEMP-20.)
        DELTAH = ABS( EL( L1 + JJ) - EL(M1))
        R = EXP (-CTR * DELTAH)
       DOSAT=14.652+(-0.41022+(0.007991-0.000077774*TEMP)*TEMP)*TEMP
       C(7,JJ) = DOSAT*(1.-R) + DOSAVE*R
C
C              C A L L    S U B R O U T I N E    'SEG'
C

  507 CALL SEG (DX1(M),Q(M0),A(M0),B(M0),EL(M0),Q(M1),A(M1),B(M1),
     *EL(M1),QL(M),C(1,M),DC(1,M),K(1,M),SINK(1,M),C(1,JJ),CP,CL(1,L),
     *PARM(1,L),SINI,NNODE(L),IT0(L),IT1(L),IT2(L),JT,T,DT,
     *DISP(M),F,NS,QWIND(L))
      C(7,JJ) = DOSAVE

  505 CONTINUE

C  "DO 505" LOOP COMPLETE - CALLS TO SEG COMPLETE FOR THIS TIME STEP

C  WRITE STATEMENTS
C
      IDAY    = ELAPSE + START + 1.
```

```
      IEHOUR = ELAPSE*24. + 0.008
      IEMIN  = (ELAPSE*24.-IEHOUR)*60. + 0.5
      ACLOCK = CLOCK*14.4 + AINT(CLOCK*24 + 0.008)*0.4


C*********************************************************************
C WRITE OUT DATA FOR PLOTTING WQ VARIABLES TO BINARY FILE 19       *
C IF IPLOT=0 PLOTTING SECTION IS SKIPPED.                          *
C*********************************************************************

      IF(IPLOT.EQ.0)GOTO 9088
      IF(J.EQ.1)WRITE(19) (RMILE(IN), IN=1,IND1)
      IF (MOD(J-1,IPLOT).EQ.0) THEN
        WRITE(19) ELAPSE
        WRITE(19) ((C( NN, IN) , NN=1,10), IN=1,IND1)
      END IF


C*********************************************************************


 9088 IF (MOD(J-1,IPRINT).NE.0) GO TO 533
C
C                  S E G M E N T    C O M P U T A T I O N

      DO 500 L = 1, NS

      M1 = NODE1(L)-1
      M2 = NNODE(L)

      WRITE(6,190) ADATE
      WRITE(6,200) TITLE
      WRITE(6,210) IEHOUR,IEMIN,ACLOCK,IDAY,ID(L),(SNAME(JJ,L),JJ=1,10)
      WRITE(6,260) (BLABEL(MM),MM=1,MAX)
      WRITE(6,270) (BUNIT(MM),MM=1,MAX)
      WRITE(6,FMT) (I,RMILE(I + M1),(C(MC(MM),I + M1),MM=1,MAX),I=1,M2)

      IF (PARM(7,L).LE.0) GO TO 500
      NDEPL0 = PARM(25,L)
      NDEPL1 = PARM(26,L)
      NOPO40 = PARM(27,L)
      NOPO41 = PARM(28,L)
  500 CONTINUE
  533 CONTINUE
C
C
C
      STB = STE
      IF (STB .GE. 24.    ) STB = STB-24.
      JDYO = JDYO
      IF (TOFDAY .GE. 24.) JDYO = JDYO + 1
      IF (TOFDAY .GE. 24.) TOFDAY = TOFDAY-24.
      IF (JDYO .GT. 365   ) JDYO = JDYO-365

  499 CONTINUE
```

```
C  "499 CONTINUE" ENDS MAIN TIME MARCH, RETURN TO "DO 499" UNTIL
C  ALL TIME STEPS COMPLETE

      READ(5,10,END = 530) TITLE
C
      DO 520 LL = 1,NS
C
C   READ CONSTANT DATA FOR EACH SEGMENT (SENSITIVITY ANALYSIS)
C
      L = IDO(LL)
      QWINDO=.FALSE.
      READ(5,CONST,END = 530)
      IF(QWINDO) QWIND(LL)=.TRUE.

      IF (LL.EQ.1) GO TO 704
      M1 = L
      M2 = L
      GO TO 705
  704 M1 = 1
      M2 = NS

  705 CONTINUE

      DO 720 M = M1,M2
C
C   PACK SEGMENT-SPECIFIC PARAMETERS INTO THE PARM ARRAY
C   AND PERFORM CONVERSIONS
C
C     CONVRT = 4.08732E-04**BK
      CONVRT     = 1.0
      PARM(1,M)  = ADN*CONVRT
      PARM(2,M)  = AG*3.79726E-05**E1*0.3048**E2
      PARM(3,M)  = AK1*CONVRT
      PARM(5,M)  = AKN*CONVRT
      PARM(6,M)  = AKNX*CONVRT
      PARM(7,M)  = ALGAEB*KALGRO
      PARM(8,M)  = ALGAEB*KALGDK
      PARM(9,M)  = APO4*CONVRT
      PARM(10,M) = ATB*CONVRT
      PARM(11,M) = ATS*CONVRT
      PARM(12,M) = BK
      PARM(13,M) = CSINK
      PARM(14,M) = E1
      PARM(15,M) = E2
      PARM(16,M) = KEXT/0.3048
      PARM(17,M) = KNCBDN
      PARM(18,M) = KOALDK
      PARM(19,M) = KOCBDN
      PARM(20,M) = KOCB1
      PARM(21,M) = KON
      PARM(22,M) = TEMP
      PARM(23,M) = TSINK
      PARM(24,M) = TSIV/0.3048
      PARM(29,M) = KCOLIDK
```

```
      PARM(30,M) = KFEDK
      PARM(31,M) = KMNDK
C
C   ASSIGN LATERAL INFLOW CONCENTRATIONS AND TEST FOR PRESENCE
C
      CL(1,M) = TEMPL
      CL(2,M) = CBODL
      CL(3,M) = ORGANL
      CL(4,M) = NH3NL
      CL(5,M) = NO3NL
      CL(6,M) = PO4L
      CL(7,M) = DOL
      CL(8,M) = MNL
      CL(9,M) = FEL
      CL(10,M) = COLIL

      DO 720 N = 1,10
      IF (CL(N,M).GT.0) CP(N) = .TRUE.
  720 CONTINUE

C
      WRITE(6,511) ICYCLE,TITLE
C
C   R E W I N D    8
C
      REWIND 8

      DO 517 I = 1,3
  517 READ(8)

  520 CONTINUE

C "520 CONTINUE" ENDS 520 LOOP, ALL RUNS NOW COMPLETE

  530 CONTINUE

C PROGRAM KICKS OUT TO "530 CONTINUE" IF LESS THAN 4 SENSITIVITY RUNS

   10 FORMAT(20A4)
   30 FORMAT(10F8.0)
  135 FORMAT('0INITIAL CONDITIONS',/,5X,10(A8))
  136 FORMAT('0BOUNDARY CONDITIONS',/,5X,10(A8))
  140 FORMAT(1X,I4,10F8.2)
  141 FORMAT(5X,10F8.2)
  142 FORMAT('0','ITERATION UPDATES FOR BOUNDARY ',I5)
  190 FORMAT('1','DYNAMIC RIVERINE WATER QUALITY MODEL',',',
     * 'CE-QUAL-RIV1', ' RIV1Q ',A9)
  200 FORMAT('0',20A4)
  210 FORMAT('0','ELAPSED TIME ',I4,' HOURS ',I2,' MINUTES',5X,'24-',
     *      'HOUR CLOCK TIME',F8.2,'   DAY ',I2,' SEGMENT',I3,' ',10A4,
     *      T61,':')
  260 FORMAT('0',T7,'RIVER',7X,2(A8,3X),A8,4X,6(A8,4X),A8)
  270 FORMAT(' ',T8,'MILE',7X,2(A8,3X),A8,4X,6(A8,4X),A8)
  481 FORMAT(' NITROGEN POOL DEPLETED FROM NODE',I3,' TO NODE ',I3)
  491 FORMAT(' PHOSPHATE POOL DEPLETED FROM NODE',I3,' TO NODE',I3)
```

```
  492 FORMAT(' NO ALGAL PRODUCTION.')
  511 FORMAT('1SENSITIVITY ANALYSIS NUMBER',I2,2X,20A4)
C
      RETURN
      END
C
C     ..........................................
C     .. S U B R O U T I N E   S E G          ..
C     ..........................................

      SUBROUTINE SEG(DX1,Q0,A0,B0,EL0,Q,A,B,EL,QL,C,DC,K,SINK,BC,
     *CP,CL,PARM,SINI,MNODE,IT0,IT1,IT2,JT,T,DT,DISP,F,NS,QWIND)

C
      IMPLICIT REAL(K)

C   COMMON BLOCKS

      COMMON/STAR/    START,          TEMP
      COMMON/TIME/    STB,            STE,            DELTAT,
     *                TOFDAY,         JDY0,           LII,
     *                LMN,            CONS7,          I,
     *                J
      COMMON/PERIOD/  LMREAD,         HSTART,         ITEM

      COMMON/SUB/     JDY0,           DUC,            LSMC,
     *                LATUDC,         LONTUC,
     *                TEM0,           WIND0,          CLOUD0,
     *                ATM0,           DEPTH0,         WSE0,
     *                RHO0,           WET0,           DRY0,
     *                HFLUX,          DTEM,           FWST,
     *                HNEFSW,         OXIDAT,
     *                KFEDK,          KMND:,          KCOLIDK

      REAL            LATUDC,         LONTUC,         LSMC
      CHARACTER       ITEM0*8,        ITEM1*8,        UNIT0*20,
     *                UNIT1*20
C
      DIMENSION DX1(MNODE),Q0(MNODE),A0(MNODE),B0(MNODE),EL0(MNODE),
     *Q(MNODE),A(MNODE),B(MNODE),EL(MNODE),QL(MNODE),C(10,MNODE),
     *DC(10,MNODE),K(10,MNODE),SINK(10,MNODE),BC(10),CP(10),PARM(31),
     *JT(NS),T(42,1),CL(10),DISP(MNODE),F(MNODE,6)

      LOGICAL     CP,       JUNCT,  QWIND
      REAL        NO3NX,    NPOOL

      DATA        THETA/0.55/

      RTHETA = (THETA - 1.)/THETA
C
C  ASSIGN PARAMETERS
C
      ADN    = PARM(1)
      AG     = PARM(2)
      AK1    = PARM(3)
```

```
      AKN    = PARM(5)
      AKNX   = PARM(6)
      ALG1   = PARM(7)
      ALG0   = PARM(8)
      APO4   = PARM(9)
      ATB    = PARM(10)
      ATS    = PARM(11)
      BK     = PARM(12)
      CSINK  = PARM(13)
      E1     = PARM(14)
      E2     = PARM(15)
      KEXT   = PARM(16)
      KNCBDN = PARM(17)
      KOALDK = PARM(18)
      KOCBDN = PARM(19)
      KOCB1  = PARM(20)
      KON    = PARM(21)
      TEQ    = PARM(22)
      TEMP   = PARM(22)
      TSINK  = PARM(23)
      TSIV   = PARM(24)
      KCOLIDK= PARM(29)
      KFEDK  = PARM(30)
      KMNDK  = PARM(31)
C
C     NEW HEAT SUBROUTINE
C

      L    = IT2 + 1
      IT   = MAX0(1,IT1)
      IF (DT.EQ.0.) GO TO 11
      ULL = Q0(MNODE)/A0(MNODE)
C
C CALCULATE THE DERIVATIVE AT THE FIRST NODE
C
      D0   = DT/2.*(Q0(1)/A0(1) + Q(1)/A(1))
      DEN = 1.5*DX1(1) + 2.*D0
      B1   = 1.5*DX1(1)/(D0*DEN)
      B2   = 3.*D0/(DX1(1)*DEN)
      B3   = D0/DEN

      DO 63 N = 1,10
        IF(CP(N)) THEN
C  RELFECT BOUNDARY CONDITIONS AT NODE 1 WHEN THERE IS NO INFLOW
C  AT THE UPSTREAM BOUNDARY. (1 CMS OR 86400 CUBIC METERS/DAY IS
C  CONSIDERED NO FLOW)
          IF(Q(1).LT.86400.) THEN
            BC(N)=C(N,2)
          END IF
      DC(N,1) = B1*(C(N,1)-BC(N)) + B2*(C(N,2)-C(N,1))-B3*DC(N,2)
        END IF
   63 CONTINUE
      GO TO 14
C
C  PLACE INITIAL CONCENTRATIONS INTO TRIBUTARY DATA TRANSFER ARRAY (T-ARRY)
```

```
C
   11 IF (IT1.EQ.0) GO TO 14

      DO 15 LL = IT1,IT2

      I = JT(LL) - 1

      DO 15 N  = 1,10
      IF (.NOT. CP(N) ) GO TO 15
      T(N + 22,LL) = C(N,I)
      T(N + 32,LL) = DC(N,I)
   15 CONTINUE
C
C    INITIALIZE NUTRIENT DEPLETION PARAMETERS
C
   14 NDEPL1 = MNODE + 1
      NOPO41 = MNODE + 1
      NDEPL0 = 0
      NOPO4  = 0
C
C                        N O D E   C O M P U T A T I O N

C
C  "DO 470" - NODE MARCH.  CERTAIN PROGRAM BLOCKS ARE BYPASSED FOR
C  DT=0 (INITIAL CONDITIONS) AND I=1 (BOUNDARY CONDITIONS).
C  LOOP MARCHES THROUGH NODES IN AN UPSTREAM DIRECTION AND
C  CYCLES THRU EVERY NODE IN THE SYSTEM ONCE DURING EACH TIME STEP.
C
      DO 470 II = 1,MNODE

      LII = II

      I = MNODE-II + 1
C
C  IN ORDER TO ELIMINATE THE TURBULENCE CALCULATION,
C  QX IS BEING SET EQUAL TO 1.0
C
C       QX  = ABS(Q(I))**BK
        QX = 1.0

        UUR = Q(I)/A(I)
        UUL = U R
        IF(I.NE.1) UUL = Q(I-1)/A(I-1)
C
C  CHECK TO SEE IF WE HAVE AN INITIAL OR BOUNDARY CONDITION
      IF (DT.EQ.0..OR.I.EQ.1) GO TO 150

C  CHECK TO SEE IF NODE IS AT A JUNCTION, IF NOT GO TO STATEMENT 12
      IF (L.LE.IT) GO TO 12
      IF (JT(L-1).NE.I) GO TO 12
C
C  CALCULATE DILUTION RATIOS FOR TRIBUTARY JUNCTIONS
C
      L = L-1
      JUNCT = .TRUE.
```

```
      D = T(21,L)/Q(I + 1)
      DD = T(22,L)*D/UUR
      GO TO 13
   12 JUNCT = .FALSE.
   13 ULL   = Q0(I-1)/A0(I-1)

      DX  = DX1(I-1)
      DUDX= (ULR-ULL)/DX
      DADX= (A0(I) - A0(I-1))/DX
      DADX2=(A(I) - A(I-1))/DX
      DDISP= (DISP(I) - DISP(I-1))/DX
      DDA1 = DISP(I)/A0(I)*DADX + DDISP
      DDA2 = DISP(I)/A(I)*DADX2 + DDISP
      RDDA = DDA1/DDA2
C   "US" IS THE CHARACTERISTIC VELOCITY OF THE CONSTITUENT CONCENTRATION
C   AND "E" IS THE RESULTING COURANT NUMBER.  THESE PARAMETER ARE COMPUTED
C   ONCE FOR EACH NODE DURING EACH TIME STEP.

      US  = (UUR + ULR-DDA1-DDA2)/(2. + DT*DUDX)
      E   = US*DT/DX
      EC  = 1.-E
      AS  = (A(I) + A0(I-1)*E + A0(I)*EC)/2.
      QLA = (QL(I)*(1. + EC) + QL(I-1)*E)/(2.*AS)
      DQLA= ((QL(I)-QL(I-1))/DX-QLA * DADX)/AS
      G0  = 1.-DT*DUDX

C    "A1" THRU "A4" ARE THE COEFFICIENTS FOR THE POLYNOMIAL ESTIMATOR OF THE
C    CONCENTRATION AT NODE I AND TIME STEP J
      A1  = E*E*(3.-2.*E)
      A2  = 1.-A1
      A3  = E*E*EC*DX
      A4  = -E*EC*EC*DX

C   "US" IS REDEFINED AS CHARACTERISTIC VELOCITY OF THE SPATIAL DERIVATIVE
C   OF THE CONSTITUENT CONCENTRATIN AND "ED" IS THE RESULTING COURANT
C   NUMBER.  THEY ARE COMPUTED ONCE FOR EACH NODE DURING EACH TIME STEP.

      US  = US - DDISP
      ED  = US * DT / DX
      EDC = 1. - ED

C    "B1 THRU B4" ARE THE COEFFICIENTS OF THE POLYNOMIAL ESTIMATOR OF THE
C    SPATIAL DERIVATIVE AT NODE I AND TIME STEP J.

      B1  = 6.*ED*(-EDC)/DX
      B2  = -B1
      B3  = ED*(ED-EDC-EDC)
      B4  = EDC*(EDC-ED-ED)
C
C    AVERAGE VALUE OF D.O., NO3, NPOOL, AND PO4
C
      DOX = C(7,I-1)*E + C(7,I)*EC
      IF (DOX.LT.0.0) DOX = 0.0
      NO3NX = C(5,I-1)*E + C(5,I)*EC
      NPOOL = C(4,I-1)*E + C(4,I)*EC + NO3NX
```

```
      PO4X  = C(6,I-1)*E + C(6,I)*EC
      GO TO 140
C
C   VALUE OF D.O., NO3, NPOOL, AND PO4 USED FOR INITIAL AND BOUNDARY CONDITIONS
C
  150 DOX  = C(7,I)
      NO3NX= C(5,I)
      NPOOL= NO3NX + C(4,I)
      PO4X = C(6,I)
C
C
C   COMPUTE HYDRAULIC DEPTH FOR CURRENT NODE
  140 H = A(I)/B(I)

C   STATEMENTS BELOW SHUT DOWN ALGAL GROWTH IN DARKNESS, (SINI.LE.0.0),
C   OR LACK OF NUTRIENTS, (PO4X OR NPOOL .LE. 0.0)

      IF (SINI.LE.0.) ALGRO = 0.0
      IF (PO4X.GT.0.0) GO TO 310
      ALGRO = 0.0
      IF (NOPO41.GT.MNODE) NOPO41 = I
      NOPO40 = I
  310 IF (NPOOL.GT.0.0) GO TO 315
      ALGRO = 0.0
      IF (NDEPL1.GT.MNODE) NDEPL1 = I
      NDEPL0 = I

C  TEMP DEFINED AS C(1,I) FOR USE IN TEMP. CORRECTION EQUATIONS BELOW

  315 IF (.NOT.CP(1))GO TO 319
      TEMP = C(1,I)

C    THE FOLLOWING THREE STATEMENTS ARE TEMPERATURE CORRECTIONS FOR
C    VARIOUS TYPES OF REACTIONS

  319 TBIO  = 1.047**(TEMP-20.)
      TNH3  = 1.1**(TEMP-20.)
      TPHYS = 1.024**(TEMP-20.)

C    DOSAT IS THE SATURATION VALUE OF D.O. AS A FUNCTION OF WATER TEMP
      DOSAT=14.652+(-0.41022+(0.007991-0.000077774*TEMP)*TEMP)*TEMP
C
C*****************************************************************************
C       ALGAE ARE LIGHT LIMITED ONLY IN THIS VERSION
C
        IF (SINI.LE.0.) GO TO 330

C       IF YOU WANT TO INCLUDE PHOSPHATE AND NITROGEN LIMITATION TO
C       ALGAL GROWTH, COMMENT OUT THE ABOVE STATEMENT AND UNCOMMENT
C       THE ONE BELOW
C
C  320 IF (SINI.LE.0..OR.PO4X.LE.0.0.OR.NPOOL.LE.0.0) GO TO 330

C*****************************************************************************
C     LIGHT LIMITATION UNDER EQUILIBRIUM TEMPERATURE APPROACH.
```

```
C     NOTE:  DAWN, SUNSET, HNEFSW MUST BE SPECIFIED IN CONSTANT LIST FOR
C     THIS CASE.
C*****************************************************************************

      IF(ITEM.EQ.1)THEN
          SWALG= HNEFSW*SINI
      ELSE
          SWALG= HNEFSW
      END IF

C  ALGAL GROWTH RATE (ALGRO) DEFINED

      IF(KEXT.GE.0.01)
     *ALGRO = ALG1*SWALG*(2./KEXT + (B(I)-2./KEXT)*EXP(-KEXT*H))/A(I)
      IF(KEXT.LT.0.01)
     *ALGRO = ALG1*SWALG*(H*(2.-KEXT*H) + B(I)*EXP(-KEXT*H))/A(I)

  330 IF (NPOOL .GT. 0.) P = NO3NX/NPOOL

C     WHEN D.O. DROPS TO ZERO ALGAL DECAY, AEROBIC HYDROLYSIS OF ORG-N,
C     AND NITRIFICATION CEASE.  THEREFORE ALGADK, K1, AND KN SET TO ZERO.

      IF (DOX.LE.1.E-06) GO TO 380
      ALGADK = ALG0 *(B(I) + H + H)/((1. + KOALDK/DOX)*A(I))
      K1     = AK1  *QX *TBIO/(1. + KOCB1/DOX)
      KN     = AKN  *QX *TNH3/(1. + KON/DOX)
      KMNX   = KMNDK
      KFEX   = KFEDK
      GO TO 390
  380 ALGADK = 0.0
      K1     = 0.0
      KN     = 0.0
  390 KDN = ADN*QX*TBIO*KOCBDN/(DOX + KOCBDN)
     *              *NO3NX/(NO3NX + KNCBDN)
      KNX = AKNX*QX*TPHYS

C    WHEN THE D.O., (DOX), LEVEL DROPS BELOW A SPECIFIED VALUE (OXIDAT)
C    IRON AND MANGANESE OXIDATION STOPS, KMNX AND KFEX SET TO 0.0.

      IF (DOX .LT. OXIDAT) KMNX = 0.
      IF (DOX .LT. OXIDAT) KFEX = 0.
C
C    THIS CODE SEGMENT (DOWN TO 391) COMPUTES REAERATION COEFFICIENTS

      K2 = AG*ABS(UUR)**E1/H**E2*TPHYS

      IF (I.EQ.1 .OR. DT.EQ.0.) GO TO 391

C    COMPUTE TSIVOGLOU REAERATION COEFFICIENT, K2, IF TSIV .GT. 0.
C    OTHERWISE USE EXPONENTIAL REAERATIOIN COEFFICIENT, K2, COMPUTED ABOVE

        IF (TSIV.GT.0.) THEN
        DELTAH = EL(I) - EL(I-1)
        K2 = TSIV * TPHYS * ABS(DELTAH)*0.5* ABS( UUR + UUL)/DX
        END IF
```

```
  391   CONTINUE

C     END REAERATION ROUTINES

      KPO4DK = APO4*QX*TPHYS
C
C  USE EQUILIBRIUM TEMPERATURE APPROACH IF ITEM=1 (IN CONSTANT LIST)
C  CONVERSION FACTOR CHANGES UNITS FROM W/M2-C TO PER DAY (0.02064=
C  86400/(4184*1000)
C
        IF(ITEM.EQ.1)THEN
          KTS=ATS*0.02064/H
          KTB=ATB*TPHYS*QX
        END IF

C
C  "DO 460" LOOP COMPUTES DECAY CONSTANTS AS WELL AS KINETIC CONSTANTS
C  FOR SOURCES OR SINKS FOR A GIVEN NODE.  CONSTITUENT CONCENTRATIONS
C  CORRECTED FOR DECAY, SOURCES OR SINKS ARE THEN COMPUTED

      DO 460 N = 1,10

      IF (.NOT.CP(N)) GO TO 460

      IF (DT.EQ.0..OR.I.EQ.1) GO TO 450

      KS     = K(N,I-1)*E + K(N,I)*EC
      DK     = (K(N,I)-K(N,I-1))/DX
      SINKS  = SINK(N,I-1)*E + SINK(N,I)*EC
      DSINK  = (SINK(N,I)-SINK(N,I-1))/DX

  450 GO TO (1,2,3,4,5,6,7,8,9,10),N
C
C     TEMPERATURE
C
C   DECAY RATE AND SOURCE/SINK TERM FOR TEMPERATURE, K(1,I),
C   INITIALLY SET TO ZERO

    1   K(1,I) = 0.
        SINK(1,I) = 0.
C

C   EQUILIBRIUM TEMPERATURE SIMULATION USED IF ITEM=1
C   FULL HEAT BALANCE SIMULATION USED IF ITEM=0
C   IF ITEM NOT EQUAL TO 0 OR 1 HEAT EXCHANGE DOES NOT OCCUR

        IF ( ITEM.NE.1 .AND. ITEM.NE.0 ) GO TO 480

C   EQUILIBRIUM APPROACH FOR COMPUTING TEMPERATURE SINK

        IF(ITEM.EQ.1)THEN
            K(1,I)    =  KTS + KTB
            SINK(1,I)=  KTS * TEQ + KTB * TSINK
        GO TO 480
        END IF
```

```
C
      DEPTH0 = H * 3.281

C   FULL HEAT BALANCE APPROACH (ITEM=0; DEFAULT VALUE) REQUIRES CALLING
C   SUBROUTINES HEAT AND HTFLUX

      IF(ITEM.EQ.0.)THEN
        CALL HEAT
C
        IF (J .NE. 1) THEN
        K(1,I) = KTB
        SINK(1,I) = DTEM/DT + KTB * TSINK
        END IF
        GO TO 480
      END IF

C  END OF TEMPERATURE COMPUTATION SEGMENT
C
C       CBOD
C
   2  K(2,I) = K1 + KDN
      SINK(2,I) = 0.0
      GO TO 480
C
C     ORGANIC NITROGEN
C
   3  K(3,I) = K1 + KDN
      SINK(3,I) = 0.0
      CP(4) = .TRUE.
      GO TO 480
C
C     AMMONIA NITROGEN - K(4,I) REPRESENTS NH3 LOSS DUE TO ADSORPTION
C     ONTO SEDIMENT AND NITRIFICATION.  SINK(4,I) REPRESENTS NET NH3
C     LOSS BY ALGAL UPTAKE AND PRODUCTION BY ALGAL DECAY AS WELL AS
C     NH3 PRODUCTION BY HYDROLYSIS OF ORG-N.
C
   4  K(4,I) = KN + KNX
      SINK(4,I) = (K1 + KDN)*C(3,I)-.075*((1.-P)*ALGRO-ALGADK)
      IF(AKN.GT.0.0)CP(5) = .TRUE.
      GO TO 480
C
C     NITRATE NITROGEN - SINK(5,I) REPRESENTS NO3 FORMATION BY
C     NITRIFICATION AND NO3 LOSS BY DENITRIFICATION AND PLANT UPTAKE
C
   5  K(5,I) =  0.0
      SINK(5,I) = KN*C(4,I)-.075*P*ALGRO-.35*KDN*C(2,I)
      GO TO 480
C
C     PHOSPHATE - K(6,I) REPRESENTS THE ADSORPTION (LOSS) OF P
C     ONTO SEDIMENTS.  SINK(6,I) REPRESENTS UPTAKE OF P BY ALGAL
C     GROWTH AND RELEASE OF P BY ALGAL DECAY.
C
   6  K(6,I) = KPO4DK
      SINK(6,I) = 0.010*(ALGADK-ALGRO)
      GO TO 480
```

```
C
C     DISSOLVED OXYGEN (DOX) - K(7,I) REPRESENTS LOSS OF DOX TO THE
C     ATMOSPHERE BY REAERATION.  SINK(7,I) REPRESENTS GAIN OF DOX BY
C     PHOTOSYNTHESIS AND REAERATION; LOSS OF DOX BY CBOD EXERTION,
C     NITRIFICATION, PLANT RESPIRATION, AND OXIDATION OF FE AND MN
C
   7  K(7,I) = K2
      SINK(7,I) = (.35*P + 1.59)*ALGRO-1.59*ALGADK + K2*DOSAT
     *-4.57*KN*C(4,I)-K1*C(2,I)-0.15*KMNX*C(8,I)-0.14*KFEX*C(9,I)
      GO TO 480
C
C     DISSOLVED MANGANESE - K(8,I) REPRESENTS MN LOSS BY OXIDATION.
C
   8  IF (C(7,I) .GT. OXIDAT) THEN
        K(8,I) = KMNDK
      ELSE
        K(8,I) = 0.0
      END IF
      SINK(8,I) = 0.0
      GO TO 480
C
C     DISSOLVED IRON - K(9,I) REPRESENTS FE LOSS BY OXIDATION
C
   9  IF (C(7,I) .GT. OXIDAT) THEN
        K(9,I) = KFEDK
      ELSE
        K(9,I) = 0.0
      END IF
      SINK(9,I) = 0.0
      GO TO 480
C
C     COLIFORM BACTERIA - K(10,I) REPRESENTS LOSS OF FECAL COLIFORMS
C
  10  K(10,I) =  KCOLIDK * TBIO
      SINK(10,I) = 0.0
      GO TO 480
C
C     STATEMENTS WHICH HANDLE ALL PARAMETERS IN THE SAME WAY
C
 480  IF (DT.EQ.0..OR.I.EQ.1 ) GO TO 460
      KS    = (K(N,I) + KS)/2.
      SINKS = (SINK(N,I) + SINKS)/2.

C
C     FINAL CALCULATION OF THE CONSTITUENT CONCENTRATION AND SPATIAL
C     DERIVATIVE FOR ALL NODES.  STATEMENT BELOW CHECKS TO SEE IF
C     NODE IS A JUNCITON POINT.  IF SO GO TO 38.
C
      IF (JUNCT) GO TO 38

      CO  = A1*C(N,I-1) + A2*C(N,I) + A3*DC(N,I-1) + A4*DC(N,I)
      CCL = CL(N)-CO
        IF(QLA.LE.0.) CCL =0.
      DC(N,I)=(B1*C(N,I-1) + B2*C(N,I) + B3*DC(N,I-1) + B4*DC(N,I))
     *    *(GO-DT*(KS + QLA)) + DT*(DSINK-CO*DK + CCL*DQLA)
```

```
C  C(N,I) IS THE CONCENTRATION OF CONSTITUENT N AT (NON-JUNCTION) NODE I,
C  TIME INCREMENT J, CORRECTED FOR SOURCES, SINKS AND LATERAL INFLOW

      C(N,I) = C0*(1-DT*KS) + (SINKS + CCL*QLA)*DT

      GO TO 465
C
C  C(N,I), BELOW, IS THE CONCENTRATION OF CONSTITUENT N AT (NON-JUNCTION)
C  NODE I, TIME INCREMENT J, CORRECTED FOR SOURCES, SINKS AND LATERAL INFLOW
C
   38 C00  = T(N + 22,L)
      DC00 = T(N + 32,L)
      C0   = A1*C(N,I-1) + A2*C00 + A3*DC(N,I-1) + A4*DC00
      CCL  = CL(N)-C0
        IF(QLA.LE.0.) CCL = 0.
      DC00 = (B1*C(N,I-1) + B2*C00 + B3*DC(N,I-1) + B4*DC00)
     *      .      *(G0-DT*(KS + QLA)) + DT*(DSINK-C0*DK + CCL*DQLA)
      C00  = C0*(1-DT*KS) + (SINKS + CCL*QLA)*DT
      T(N + 22,L) = C00
      T(N + 32,L) = DC00
      C(N,I)  = C00*(1.-D) + T(N,L)*D
      DC(N,I) = DC00*(1.-D) + T(N + 10,L)*DD

  465 CONTINUE

C ***************************************************************************
C       CALL TO WIND-DRIVEN REAERATION SUBROUTINE REAERK                    *
C                                                                           *
C       IF ITEM=1, TAIR=DRY0 AND WIN=WIND0*0.447 WITH THE ACTUAL VALUES     *
C       PUT IN FOR DRY0 AND WIND0 BY USER.                                  *
C       IF ITEM=0, TAIR=DRY0 AND WIN=WIND0*0.447, VALUE FOR TAIR AND        *
C       WIN ARE OBTAINED FROM MET DATA DURING HEAT EXCHANGE CALCULATION     *
C***************************************************************************
        IF (N.EQ.7 .AND. QWIND)  THEN
                  TAIR=DRY0
                  WIN=WIND0*0.447
         IF(ITEM.EQ.1.AND.TAIR.EQ.0.AND.WIN.EQ.0)THEN
            WRITE(6,9089)
 9089 FORMAT(' USER MUST PROVIDE A CONSTANT VALUE FOR DRY BULB TEMP',/,
     * ' AND WIND SPEED (TAIR AND WIN) SINCE ITEM=1 AND QWIND=TRUE.',/,
     * ' DO THIS BY REPLACING DRY0 AND WIND0 WITH NUMERICAL VALUES',/,
     * ' IN THE STATEMENTS DIRECTLY ABOVE LINE 9089 IN THE SOURCE',/,
     * ' CODE.  IF NOT DONE DRY0 AND WIND0 ARE SET TO 0 AND THIS ',/,
     * ' ERROR RESULTS.')
            STOP
            END IF
                    CALL REAERK(WIN,TAIR,TEMP,RK)
                    DTS = DT * 86400.
                    RK = RK/86400.
                    C(7,I) = C(7,I) + RK * (DOSAT-C(7,I)) * DTS / H
                END IF

C ***************************************************************************
C       NEGATIVE HEDGE FOR ALL WQ CONSTITUENTS EXCEPT TEMPERATURE.
```

```
C       THAT IS, NO CONCENTRATIONS ARE ALLOWED TO GO BELOW ZERO.
C ***************************************************************
        IF (N.GT.1) THEN
                IF (C(N,I).LT.0.) C(N,I) = 0.
        END IF

  460 CONTINUE

C ........  END "DO 460" CONCENTRATION CONSTITUENT LOOP ........

C     SHIFT POSITIONS
      ULR=ULL

  470 CONTINUE
C ........  END OF "DO 470" NODE COMPUTATION LOOP ..............

      IF (ITO .LE. 0 ) GO TO 560
C
C   VALUES AT THE DOWNSTREAM TERMINUS PLACED IN THE T-ARRAY
C
      T(21,ITO) = Q(MNODE)
      T(22,ITO) = Q(MNODE)/A(MNODE)

      DO 16 N = 1,10
      IF (.NOT.CP(N)) GO TO 16
      T(N,ITO) = C(N,MNODE)
      T(N + 10,ITO) = DC(N,MNODE)
   16 CONTINUE

 560  PARM(25) = NDEPLO
      PARM(26) = NDEPL1
      PARM(27) = NOPO40
      PARM(28) = NOPO41
C
C  ASSIGN BOUNDARY CONDITIONS TO THE FIRST NODE
C
      IF (DT.LE.0.) RETURN
      DO 403 N = 1,10
      IF (CP(N)) C(N,1) = BC(N)
  403 CONTINUE
C
C  AT THIS POINT, THE COMPUTED CONSTITUENT CONCENTRATIONS ARE CORRECTED
C  FOR DISPERSION USING AN IMPLICIT NUMERICAL SCHEME
C
      F(1,2) = 1.
      F(1,3) = 0.
      F(MNODE,1) = 0.
      F(MNODE,2) = 1.
      MN1 = MNODE - 1
      DO 500 I = 2,MN1
      FO = 2.*DISP(I)*THETA*DT/(DX1(I-1) + DX1(I))
      F(I,1) = -FO/DX1(I-1)
      F(I,3) = -FO/DX1(I)
      F(I,2) = 1. - F(I,1) - F(I,3)
500   CONTINUE
```

```
C
C  NOW THE RIGHT-HAND SIDE FOR EACH MODELED CONSTITUENT
C
      DO 510 N = 1,10
      IF(.NOT.CP(N))GO TO 510
      F(1,4) = C(N,1)
      F(MNODE,4) = C(N,MNODE)
      DO 520 I = 2,MN1
      F(I,4) = RTHETA*(F(I,1)*C(N,I-1) + F(I,3)*C(N,I+1) +
     *  (F(I,2)-1.)*C(N,I)) + C(N,I)
520   CONTINUE

       CALL TRIDAG(F(1,1),F(1,2),F(1,3),F(1,4),F(1,5),F(1,6),
     * C,MNODE,N)
       F(1,4) = DC(N,1)
       F(MNODE,4) = DC(N,MNODE)
       DO 540 I = 2,MN1
       F(I,4) = RTHETA*(F(I,1)*DC(N,I-1) + F(I,3)*DC(N,I+1) +
     *  (F(I,2)-1.)*DC(N,I)) + DC(N,I)
540    CONTINUE

       CALL TRIDAG(F(1,1),F(1,2),F(1,3),F(1,4),F(1,5),F(1,6),
     * DC,MNODE,N)
510   CONTINUE
C
C  DISPERSION FOR TRIBUTARY JUNCTIONS
C
      IF (IT1.LE.0) RETURN
      DO 550 LL = IT1,IT2
      I = JT(LL)
      F0 = 2.*DISP(I)*DT/(DX1(I-1) + DX1(I))
      F1 = F0/DX1(I-1)
      F3 = F0/DX1(I)
      F2 = 1. + F1 + F3

      DO 550 N=1,10
      IF (.NOT.CP(N)) GO TO 550
      T(N + 22,LL) = (T(N + 22,LL) + F1*C(N,I-1) + F3*C(N,I+1))/F2
      T(N + 32,LL) = (T(N + 32,LL) + F1*DC(N,I-1) + F3*DC(N,I+1))/F2
550   CONTINUE

      RETURN
      END
C
C     ..................................................
C     ..    S U B R O U T I N E    S P L I N E       ..
C     ..................................................


C   SPLINE: A PROGRAM TO FIT A CUBIC SPLINE TO THE DATA, WITH
C   CONTINUOUS ZEROETH, FIRST AND SECOND DERIVATIVES.  THE OUTPUT
C   IS THE VALUE OF DERIVATIVE AT EACH POINT.

      SUBROUTINE SPLINE (C,DC,CP,DX,F,MNODE)
```

```
      DIMENSION C(10,MNODE),DC(10,MNODE),CP(10),DX(MNODE),F(MNODE,6)

      LOGICAL CP

      F(1,1)  = 0.
      F(1,2)  = 1.
      F(1,3)  = 0.5
      F(MNODE,1) = 0.5
      F(MNODE,2) = 1.
      F(MNODE,3) = 0.
      M2 = MNODE-1
      DO 2 I = 2,M2
      F(I,1) = DX(I)
      F(I,2) = 2.*(DX(I-1) + DX(I))
      F(I,3) = DX(I-1)

    2 CONTINUE

      DO 4 N = 1,10
      IF (.NOT.CP(N)) GO TO 4
      F(1,4) = 1.5/DX(1)*(C(N,2)-C(N,1))
      F(MNODE,4) = 1.5/DX(M2)*(C(N,MNODE)-C(N,M2))

      DO 3 I = 2, M2
      F(I,4) = 3.*(DX(I-1)/DX(I)*(C(N,I + 1)-C(N,I))
     *          + DX(I)/DX(I-1)*(C(N,I)-C(N,I-1)))
    3 CONTINUE
C
C              C A L L   S U B R O U T I N E   T R I D A G
C
      CALL TRIDAG (F(1,1),F(1,2),F(1,3),F(1,4),F(1,5),F(1,6),
     *             DC,MNODE,N)
    4 CONTINUE

      RETURN
      END

C     ..................................................................
C     ..        S U B R O U T I N E   T R I D A G                     ..
C     ..................................................................

      SUBROUTINE TRIDAG (A,B,C,D,BETA,GAMMA,V,M1,N)

      DIMENSION A(M1),B(M1),C(M1),D(M1),BETA(M1),
     *          GAMMA(M1),V(10,M1)

      BETA(1) = B(1)
      GAMMA(1)= D(1)/BETA(1)

      DO 1 I  = 2, M1
      BETA(I) = B(I)-A(I)*C(I-1)/BETA(I-1)
    1 GAMMA(I)= (D(I)-A(I)*GAMMA(I-1))/BETA(I)

      V(N,M1) = GAMMA(M1)
      M2 = M1-1
```

```
      DO 2 K = 1,M2
      I = M1-K
    2 V(N,I) = GAMMA(I)-C(I)*V(N,I + 1)/BETA(I)

      RETURN
      END
C
C     ..................................................
C     ..       S U B R O U T I N E     H E A T       ..
C     ..................................................
C
C   THIS SUBROUTINE IS ADAPTED FROM "QUAL-II" STREAM
C   QUALITY MODEL, WRITTEN BY LARRY A. ROESNER, PAUL
C   R. GIGUERE AND DONALD E. EVENSON IN WATER RESOURCES
C   ENGINEERS, INC.  710 SOUTH BROADWAY, WALNUT CREEK,
C   CALIFORNIA 94596
C

      SUBROUTINE HEAT

C   COMMON BLOCKS
C ***************************************************************************
C * METEOROLOGICAL INPUT CURRENTLY LIMITED TO 300 UPDATES AND 5 VARIABLES *
C ***************************************************************************
      DIMENSION NUP(300), DUMMY(300,5)
      COMMON/STAR/    START,          TEMP
      COMMON/TIME/    STB,            STE,            DELTAT,
     *                TOFDAY,         JDYO,           LII,
     *                LMN,            CONS7,          I,
     *                J
      COMMON/PERIOD/ LMREAD,          HSTART,         ITEM

      COMMON/SUB/     JDYO,           DUC,            LSMC,
     *                LATUDC,         LONTUC,
     *                TEMO,           WINDO,          CLOUDO,
     *                ATMO,           DEPTHO,         WSEO,
     *                RHOO,           WETO,           DRYO,
     *                HFLUX,          DTEM,           FWST,
     *                HNEFSW,         OXIDAT,
     *                KFEDK,          KMNDK,          KCOLIDK
      REAL            LATUDC,         LONTUC,         LSMC
      CHARACTER       ITEMO*8,        ITEM1*8,        UNITO*20,
     *                UNIT1*20
C
C   DEFAULT VALUE
C
      DATA            DEPTHO/5.0/,    RHOO/1.0/

C
C   IRFLAG : 0, READ ONLY ONCE
C            1, NO LONGER READ DATA
C   JDYO   : THE DAY OF YEAR TO START SIMULATION
C   LMN    : COUNTER FOR SEGMENT NUMBER
C   LII    : COUNTER FOR NODE NUMBER
```

```
C    LATUDC : LOCAL LATITUDE, DEGREE
C    LONTUC : LOCAL LONGITUDE, DEGREE
C    LSMC   : LONGITUDE OF STANDARD MERIDIAN, DEGREE
C    DUC    : DUST ATTENUATION COEFFICIENT
C    NUM    : NUMBER OF METEOROLOGICAL UPDATES
C    NUP(I) : LARGEST ITERATION AT WHICH DATA I APPLIES
C    DUMMY  : ARRAY CONTAINING ALL METEOROLOGICAL UPDATES
C    CLOUD0 : CLOUD COVER (FRACTION)
C    WIND0  : WIND SPEED (MILE/HR)
C    DRY0   : DRY BULB TEMPERATURE (F)
C    WET0   : WET BULB TEMPERATURE (F)
C    ATM0   : ATMOSPHERIC PRESSURE (INCH/HG)
C
C    THE FOLLOWING DATA ARE ONLY READ ONCE
C    IRFLAG WILL EQUAL 0 UNLESS SPECIFIED OTHERWISE
C    AS IS DONE ELSEWHERE
C

      IF (IRFLAG .EQ. 0) THEN
       WRITE(16,91) IRFLAG
        READ(02,  94)        JDY0
        READ(02,  95)        DUC
        READ(02,  95)        LATUDC, LONTUC, LSMC
        READ(02,96)NUM
        WRITE(16,90) JDY0,DUC,LATUDC,LONTUC,LSMC,NUM
        READ(02,96)(NUP(II),II=1,NUM)
        DO II=1,NUM
            READ(02,97)(DUMMY(II,III),III=1,5)
            WRITE(16,89) II,NUP(II)
            WRITE(16,88) DUMMY(II,1),DUMMY(II,2),DUMMY(II,3),
     *        DUMMY(II,4),DUMMY(II,5)
        END DO
        IRFLAG = 1

        CLOUD0 = DUMMY(1,1)
        WIND0  = DUMMY(1,2)
        DRY0 =   DUMMY(1,3)
        WET0 =   DUMMY(1,4)
        ATM0=    DUMMY(1,5)
C
        END IF

C
C
C    R E A D   L O C A L   C L I M A T O L O G I C A L   D A T A
C
       IF (J .GT. NUP(IRFLAG).AND.LMN.EQ.1 AND.LII.EQ.1)THEN
        IRFLAG = IRFLAG + 1
        CLOUD0 = DUMMY(IRFLAG,1)
        WIND0  = DUMMY(IRFLAG,2)
        DRY0 =   DUMMY(IRFLAG,3)
        WET0 =   DUMMY(IRFLAG,4)
        ATM0=    DUMMY(IRFLAG,5)
       END IF
```

```
      FWST = TEMP

   91   FORMAT(/1X,'ECHO OF MET DATA',3X,'IRFLAG=',I3)
   90   FORMAT(/1X,'JDY0=',I3,1X,'DUC=',F5.2,1X,'LATUDC=',F6.2,1X,
     *  'LONTUC=',F6.2,1X,'LSMC=',F6.2,1X,'NUM=',I3)
   89   FORMAT(1X,'BOUNDARY CHANGE',1X,I3,3X,'AT TIME STEP',1X,I3)
   88   FORMAT(1X,5(F5.2,2X),/)


C   CALL TO SUBROUTINE HEATFLUX
C
      CALL HEATFLUX (JDY0,              DUC,              LSMC,
     *               LATUDC,           LONTUC,
     *               TEMP,             WIND0,            CLOUD0,
     *               ATM0,             DEPTH0,           WSE0,
     *               RH00,             WET0,             DRY0,
     *               HFLUX )
C
C  CONVERSION HEAT FLUX FROM 'BTU' TO KCAL, THEN TO DEGREE  (F),
C  THEN TO (C).  THUS, DTEM IS IN DEG C.
C                NOTE: HERE  AND IN CONS7
C            CP = 1.0 CAL/G-C     RHO = 1 G/CM^3
C            BTU * 0.252 = KCAL
C            28.33 = (100CM)**3/(1000 CAL/KCAL* 35.31 FT**3/M**3)
C
      DTEM  = (HFLUX * 0.252) /CONS7
      FWST  = FWST + DTEM
      TEMP  = FWST

   92   FORMAT(A8,F8.2,3X,A8)
   93   FORMAT(5X,A8,F8.2,3X,A8)
   94   FORMAT(8X,8I8)
   95   FORMAT(8X,6F8.2)
   96   FORMAT(10I8)
   97   FORMAT(8X,5F8.0)
      RETURN
      END

C   ..................................................
C   ..  S U B R O U T I N E   H E A T F L U X      ..
C   ..................................................
C
C   ..............................................
C   .. THIS IS GENERAL SUBROUTINE TO CALCULATE ..
C   .. THE HEAT.  THE FOLLOWING DATA HAVE TO   ..
C   .. BE SUPPLIED, EITHER BY READING FROM     ..
C   .. OWN, OR FROM OTHER PROGRAM.             ..
C   ..                                         ..
C   ..    JDY     : DAY OF YEAR STARTING SIMU  ..
C   ..    LSM     : LONGITUDE OF STANDARD      ..
C   ..              MERIDIAN,  DEGREE          ..
C   ..    LATUD   : LOCAL LATITUDE, DEGREE     ..
C   ..    LONTUD  : LOCAL LOGITUDE, DEGREE     ..
C   ..            -1 FOR WEST,  + 1 FOR EAST   ..
C   ..    CLOUD   : CLOUDINESS, 0 - 0.99       ..
```

```
C   ..     DU       : DUST ATTENUATION, VALUE     ..
C   ..                FROM 0 - 0.13               ..
C   ..     WETBLB   : WET BULB TEMP, F            ..
C   ..     DRYBLB   : DRY BULB TEMP, F            ..
C   ..     WINSPD   : WIND SPEED, MILE/HOUR       ..
C   ..     ELEV     : WATER SURFACE ELEV,FT       ..
C   ..     TEM      : TEMPERATURE,  C             ..
C   ..     RHO      : DENSITY, GRAM PER CUBIC     ..
C   ..                MILLIMETER                  ..
C   ..     HNEF     : HEAT FLUX, VALUE RETURNED   ..
C   ..                AS BTU/FT**2                ..
C   ..                                            ..
C   ..............................................

      SUBROUTINE HEATFLUX (JDY,          DU,              LSM,
     *                     LATUD,       LONTUD,
     *                     TEM,          WINSPD,         CLOUD,
     *                     ATMPR,        DEPTH,          ELEV,
     *                     RHO,          WETBLB,         DRYBLB,
     *                     HNEF)

      COMMON/TIME/       STB,           STE,            DELTAT,
     *                   TOFDAY,        JDYO,           LII,
     *                   LMN,           CONS7,          I,
     *                   J

      COMMON/PERIOD/     LMREAD,        HSTART,         ITEM
      COMMON/SUB/        JDYO,          DUC,            LSMC,
     *                   LATUDC,       LONTUC,
     *                   TEMO,          WINDO,          CLOUDO,
     *                   ATMO,          DEPTHO,         WSEO,
     *                   RHOO,          WETO,           DRYO,
     *                   HFLUX,         DTEM,           FWST,
     *                   HNEFSW,       OXIDAT,
     *                   KFEDK,         KMNDK,          KCOLIDK
      REAL               LATUDC,        LONTUC,         LSMC
      CHARACTER          ITEMO*8,       ITEM1*8,        UNITO*20,
     *                   UNIT1*20
      REAL               LONTUD,        LATUD,          LSM
C
C   CONSTANT
C
      DATA               PI/3.141618/,  HCAP/1.0/,      RWSAR/0.03/,
     *                   HSC/438./      SPWEVA/62.22/

      ELEVP = EXP(-ELEV /2532.)

      CONS1 = 2.*PI/365.
      CONS2 = PI*LATUD  /180.
      CONS3 = 180./PI
      CONS4 = 23.45*PI/180.
      CONS5 = PI/12.
      CONS6 = 12./PI
```

```
      CONS7 = RHO*28.336*HCAP*DEPTH
      CONST3 = 6.8E-04
      CONST4 = 2.7E-04

C   STEFAN-BOLTZMAN CONSTANT

      STBOLC = 1.73E-09
C
C                    C O M P U T A T I O N
C
C           R A D I A T I O N    E X C H A N G E    H E A T
C
C  (A) NET SHORT-WAVE SOLAR RADIATION (HNSWR)
C
C   A-I.  RADIATION FLUX

      WST  = (TEM * 9.)/5. + 32.
C
C  FOR THE SAME DAY, FOLLOWING CALCULATION IS SAME
C


1234    FORMAT(1X,I4,5X,I4)
      IF ( (J .EQ. 1 .AND. LMN .EQ. 1 .AND. LII .EQ. 1) .OR.
     *     (JDY   .NE. JDYO .AND. LMN .EQ. 1 .AND. LII .EQ. 1) ) THEN
C
C  DECLINATION
C
      DESUN  = CONS4*COS(CONS1*(173.-JDY))
      DECLIN = ABS(DESUN)

C  EARTH-SUN DISTANCE

C     .17 CHANGED TO .017 IN EQUATION BELOW (3-07-89)

      REARTH = 1.0 + 0.017*COS(CONS1*(186.-JDY))

      ETIME  = 0.000121-0.12319*SIN(CONS1*(JDY-1.)-0.0714)-
     *         0.16549*SIN(2.*CONS1*(JDY-1.) + 0.3088)
C
C  DIFFERENCE BETWEEN STANDARD AND LOCAL CIVIC TIME
C
      DSTLT = (LSM - LONTUD)/15.
C
C  STANDARD TIME OF SUNRISE AND SUNSET
C
      ACS = TAN(CONS2)*TAN(DECLIN)
C
      IF (ACS .NE. 0.) THEN

      X1  = SQRT(1.-ACS*ACS)
      X1  = X1/ACS
      ACS = ATAN(X1)
      IF (DESUN .GT. 0.) ACS = PI-ACS
```

```
      ELSE
      ACS = PI/2.0

      END IF
C
      TIMRIS = 12.-(CONS6*ACS) + DSTLT
      TIMSET = 24.-TIMRIS  + 2.*DSTLT
C
      END IF
C
C   SATURATED VAPOR PRESSURE OF THE AIR, SVPA
C

      SVPA  = 0.1001*EXP(0.03*WST )-0.0837
C
C    THE FOLLOWING ARE THE SAME FOR EACH NODE BUT CAN
C    CHANGE AT EACH TIME STEP.  LMN AND LII EQUAL 1 AT
C    EACH NEW TIME STEP.
**********************************************
C
      IF ( LMN .NE. 1 .OR. LII .NE. 1 ) GO TO 480

**********************************************************************
**********************************************************************
C  IF YOU WANT TO READ IN DEW POINT TEMPERATURES INSTEAD OF WET BULB,
C  COMMENT OUT ALL STATEMENTS BETWEEN THE XXXXXXX SECTION
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
C
C.... SATURATED VAPOR PRESSURE,  SVAP
C
       SVAP  = 0.1001*EXP(0.03*WETBLB )-0.0837
C
C.... WATER VAPOR PRESSURE (WVAP)
C
       WVAP  = SVAP-0.000367*ATMPR     *
     *(DRYBLB -WETBLB )*(1.0 + (WETBLB -32.)/1571.)
C
C.... DEW POINT
C
       DEWPON  = LOG((WVAP + 0.0837)/0.1001)/0.03
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

C**********************************************************************
C  IF YOU WANT TO READ IN DEW POINT TEMPERATURES INSTEAD OF WET BULB, *
C  UNCOMMENT THE NEXT TWO STATEMENTS                                  *
C       DEWPON = WETBLB                                               *
C       WVAP = 0.1001 * EXP(0.03 * DEWPON) - 0.0837                   *
C**********************************************************************
C
C   HOUR ANGLE
C
      IF (TIMSET  .LE. STB .OR.  TIMRIS   .GE. STE)
     * THEN

      GO TO 35
```

```
      ELSE IF (TIMRIS  .GT. STB .AND. TIMRIS  .LT. STE)
     * THEN

      DIUEXP  = 1.
      ANGRIS  = TIMRIS -DSTLT  + ETIME-12.
      ANGSET  = STE    -DSTLT  + ETIME-12.

      ELSE IF (TIMSET   .LT. STE .AND. TIMSET .GT. STB)
     * THEN

      DIUEXP  = 1.
      ANGRIS  = STB-DSTLT  + ETIME-12.
      ANGSET  = TIMSET  -DSTLT   + ETIME-12.

      ELSE

      DIUEXP   = 1.
      ANGRIS   = STB-  DSTLT   +  ETIME-12.
      ANGSET   = STE-  DSTLT   +  ETIME-12.
C
      END IF
C
C  CALCULATE HOUR ANGLE,T,(HOURAN)
C
      HOURAN = (ANGRIS   + ANGSET  )/2.
C
C  SHORT-WAVE SOLAR RADIATION FLUX
C
      HRF  = HSC/(REARTH**2.)*(SIN(CONS2)*SIN(DESUN)*
     *          (ANGSET-ANGRIS) + CONS6*COS(CONS2)*
     *           COS(DESUN)*(SIN(CONS5*ANGSET)-
     *           SIN(CONS5*ANGRIS)))*DIUEXP


C
C    A-II.  ATMOSPHERIC TRANSMISSION TERM, AT(ATT)
C
C    SUN ALTITUDE IN RADIANS
C
      ALPH  = SIN(CONS2)*SIN(DESUN) +
     *COS(CONS2)*COS(DESUN)*COS(CONS5*HOURAN  )

      IF (ABS(ALPH ) .NE. 1.0) THEN

      Y1 = SQRT(1.-ALPH **2)
      Y1 = ALPH /Y1
      ALPH  = ATAN(Y1)

      ELSE IF (ALPH  .NE. -1.0) THEN
      ALPH  = PI/2.0

      ELSE
      ALPH = -PI/2.0
```

```
      END IF

      IF (ALPH  .LT. 0.01) GO TO 35
C
C   OPTICAL AIR MASS
C
      OAM  = ELEVP /(SIN(ALPH ) + 0.15*((180.*ALPH /
     *        PI + 3.885)**(-1.253)))
C
C   MEAN DAILY PRECIPITATE WATER CONTENT
C
      PWC  = 0.00614*EXP(0.0489*DEWPON )
C
C   MEAN ATMOSPHERIC COEFFICIENT, A'(AC)

      AC  = EXP(-(0.465 + 0.0408*PWC )*(0.129 + 0.171*EXP(-0.880*
     *          OAM ))*OAM )
C
C   MEAN ATMOSPHERIC TRANSMISSION COEFFICIENT
C
      ATC  = EXP(-(0.465 + 0.0408*PWC )*
     *        (0.179 + 0.421*EXP(-0.721*OAM ))*OAM )
C
C    DETERMINE THE VALUE OF AR (A), AND BR (B),
C    FROM THE VALUE OF CLOUD      FOR THE FUNCTION OF
C    REFLECTION  COEFFICIENT ,RS.
C
      CNL = CLOUD *10 + 1.0
      NLL = CNL
      GO TO (50,51,51,51,51,51,52,52,52,52,53),NLL
 50   AR = 1.18
      BR = -0.77
      GO TO 54
 51   AR = 2.20
      BR = -0.97
      GO TO 54
 52   AR = 0.95
      BR = -0.75
      GO TO 54
 53   AR = 0.35
      BR = -0.45
 54   CONTINUE
C
C  REFLECTION COEFFICIENT
C
C  SOLAR ALTITUDE IN DEGREE
C
      RS  = AR*(CONS3*ALPH )**BR
C
C  DAMPENING EFFECT ON THE SOLAR RADIATION FLUX
C  GIVES SATISFACTORY RESULTS EXCEPT FOR HEAVY
C  OVERCAST CONDITION,I.E. WHEN CLOUD
C  APPROACHES 1.0
C
      IF (CLOUD  .LE. 0.9) THEN
```

```
      CS = 1.0-0.65*CLOUD **2.
      ELSE
      CS = 0.5
      END IF
C
C   CALCULATE ATMOSPHERIC TRANSMISSION TERM
C
      ATT  = (ATC  + 0.5*(1.-AC-DU))/
     *           (1.-0.5*RS *(1.-AC   + DU))
C
C  NET SHORT-WAVE SOLAR RADIATION BTU/FT**2
C
      HNSWR   = HRF*ATT*(1.-RS)*CS

      GO TO 479

 35   DIUEXP = 0.
      HNSWR  = 0.

 479  CONTINUE
C
C  (B) CALCULATE LONG-WAVE ATMOSPHERIC RADIATION,HAN (HNLWR)
C
      HNLWR  = (2.89E-06)*STBOLC*((DRYBLB   + 460.)**6)*
     *             (1.0 + 0.17*(CLOUD  **2.))*(1.-RWSAR)

C
C  COMPUTE INSOLATION FOR ALGAE
C
      IF ( DELTAT .LE. 0. ) GO TO 481
      HNEFSW = HNSWR / DELTAT

      GO TO 482
 481  CONTINUE
      HNEFSW = 0.
 482  CONTINUE
C
C  CONVERT TO W/M**2
C
      HNEFSW = HNEFSW * 3.155
 480  CONTINUE
C
C  (C) CALCULATE WATER SURFACE BACK RADIATION, (HOLWBR)
C
       HOLWBR  = 0.97*STBOLC*((WST  + 460.)**4)
C
C                  E V A P O R A T I O N      H E A T
C
C  (D) CALCULATE EVAPORATION  (HEVAP)
C
C  D-I.  CALCULATE EVAPORATE RATE, E, (EVARAT)
C
      EVARAT  = (CONST3 + CONST4*WINSPD )*(SVPA  -WVAP )
      IF (EVARAT  .LT. 0.) EVARAT =0.
C
```

```
C  D-II. CALCULATE SENSIBLE HEAT LOSS(HSENH)
C
      HSENH  = 0.
C
C  D-III. CALCULATE LATENT HEAT OF EVAPORIZATION, HL,(HLEVA)
C
      HLEVA = 1084.-0.5*WST
C
C  HEAT LOSS BY EVAPORATION HE (HEVAP)
C
      HEVAP  = SPWEVA*HLEVA *EVARAT  + HSENH
C
C                    C O N D U C T I O N      H E A T
C
C  (E) CALCULATE CONDUCTION HEAT, HC, (HCOND)
C
      HCOND= SPWEVA*HLEVA*(CONST3 + CONST4*
     *             WINSPD)*(0.01*ATMPR /29.92)*
     *            (DRYBLB -WST)
C
C      FINALLY, CALCULATE NET ENERGY FLUX PASSING
C      THE AIR-WATER INTERFACE, HN(HNEF),BTU/FT**2,DELTAT=HOURS
C
      HNEF   = HNSWR    + (HNLWR  -HOLWBR    + HCOND  -
     *        HEVAP  )  * DELTAT

C
C  RETURN VALUE HERE
C
      RETURN
      END

      FUNCTION TAN (X)
      TAN=SIN(X/57.3)/COS(X/57.3)
      RETURN
      END

      FUNCTION ASIN(X)
      ASIN=X + (X**3./6.) + (3.*X**5./40.) + (15.*X**7./336.)
      RETURN
      END

      FUNCTION ACOS(X)
      ACOS=ASIN(SQRT(1.-X**2.))
      RETURN
      END
C       S U B R O U T I N E    R E A E R K
C*********************************************************************
C* SUBROUTINE REAERK CALCULATES REAERATION COEFFICIENT (RK) (M/DAY)
C* GIVEN WIND SPEED (WS) (M/S) AND TEMPERATURE (T) (C) USING THE
C* METHOD PRESENTED IN JOUR OF ENV ENG,VOL. 109, NO.3,PP.731-752,
C* JUNE 1983,AUTHOR: D.J.O'CONNOR, TITLE: "WIND EFFECTS ON GAS-
C* LIQUID TRANSFER COEFFICIENTS
C* PARAMETERS USED IN THIS SUBROUTINE WERE THOSE GIVEN FOR
C* INTERMEDIATE SCALE CORRELATION (TABLE 2), I.E., LARGE LABORATORY
```

```
C* SYSTEMS AND MODERATELY SIZED LAKES.
C*********************************************************************
      SUBROUTINE REAERK(WS,TA,TW,RK)
      REAL*4 KA,LAM,KA3
C
C CALCULATE DIFFUSIVITY OF OXYGEN IN WATER (DIFF) (CM**2/SEC), VISCOSITY
C OF WATER (VW) (CM**2/SEC),VISCOSITY OF AIR (VA) (CM**2/SEC),DENSITY
C OF WATER (PW) (G/CM**3), DENSITY OF AIR (PA) (G/CM**3)
C
      DIFF=4.58E-07*TW+1.2E-05
        TA = (TA-32.)*5./9.
      VW=0.0164-.00024514*TW
      VA=.133+.0009*TA
      PA=.00129-.0000040*TA
      PW=1.00
      WS=WS*100.
      RK=1.
      N=0
C USE NEWTON RAPHSON METHOD TO CALCULATE THE SQUARE ROOT OF THE DRAG
C COEFFICIENT.  PARAMETERS USED IN THE MODEL INCLUDE TRANSITIONAL
C SHEAR VELOCITY - UT(M/SEC); CRITICAL SHEAR VELOCITY- UC(CM/SEC);
C VON KARMAN'S CONSTANT-(KA); EQUILIBRIUM ROUGHNESS-ZE(CM);
C 1/LAM IS A REYNOLDS NUMBER; GAM IS A NON-DIMENSIONAL COEFFICIENT
C DEPENDENT ON THE WATER BODY SIZE.  LAM, GAM, UT, MC, AND ZE ARE
C ALSO DEPENDENT ON WATER BODY SIZE.
      UT=10.0
      UC=11.0
      KA=0.4
      KA3=KA**.3333
      ZE=0.25
      LAM=3.0
      GAM=6.5
      WH=1000.
C MAKE INITIAL GUESS FOR SQUARE ROOT OF THE DRAG COEFFICIENT
      SRCD=0.04
  10  N=N+1
C CALCULATE VALUE OF FUNCTION(F2) AND DERIVATIVE OF FUNCTION(FP)
      EF=EXP(-SRCD*WS/UT)
      F1=LOG((WH/ZE)+(WH*LAM/VA)*SRCD*WS*EF)
      F2=F1-KA/SRCD
      FP1=1./((WH/ZE)+(LAM*WH/VA)*SRCD*WS*EF)
      FP2=((WH*LAM)/(VA*UT))*SRCD*WS**2*EF
      FP3=(WH*LAM/VA)*WS*EF
      FP4=FP1*(FP2+FP3)+(KA/(SRCD**2))

C CALCULATE NEW VALUE OF SQUARE ROOT OF DRAG COEFFICIENT. COMPARE
C TO PREVIOUS VALUE. IF NOT ACCEPTABLE RETURN TO NEWTON-RAPHSON
C ALGORITHM.

      SRCD2=SRCD-F2/FP4
      ERR=ABS(SRCD-SRCD2)
      IF(ERR.GT.0.0005.AND.N.LT.8)THEN
            SRCD=SRCD2
            GO TO 10
      END IF
```

```
      IF(ERR.GT.0.005.AND.N.GE.8) GO TO 90
      CD=SRCD**2
      US=SRCD*WS
      Z0=1./((1./ZE)+LAM*US*EXP(-US/UT)/VA)
      WS=WS/100.
      IF(WS.LT.6.0)GO TO 60
      IF(WS.GE.6.0.AND.WS.LE.20.0)GO TO 80
      IF(WS.GT.20.0)GO TO 70

C CALCULATE RK FOR WINDSPEEDS LESS THAN 6.0 M/SEC BASED ON
C EQUATION 23b IN CITATION

  60  RK1=(DIFF/VW)**0.666667*SRCD*(PA/PW)**0.5
      RK=RK1*KA3*WS/GAM
      RK=RK*3600.*24.
      GO TO 85

C CALCULATE RK FOR WINDSPEEDS GREATER THAN 20 M/S BASED ON
C EQUATION 25b IN CITATION

  70  RK=(DIFF*PA*VA*US/(0.1*PW*VW))**0.5
      RK=RK*3600.*24./100.
      GO TO 85
C CALCULATE RK FOR WINDSPEEDS BETWEEN 6 AND 20 M/S BASED ON
C EQUATIONS 26a and 27 IN CITATION

  80  GAMU=GAM*US*EXP(-US/UC+1.)/UC
      RK1=(DIFF/VW)**.6667*KA3*(PA/PW)**0.5*US/GAMU
      RK2=(DIFF*US*PA*VA/(KA*Z0*PW*VW))**0.5
      RK3=(1./RK1)+(1./RK2)
      RK=1./RK3
      RK=RK*3600.*24./100.
      GO TO 85
  90  WRITE(6,102)
 102  FORMAT(5X,'SOLUTION DID NOT CONVERGE')
  85  CONTINUE
      RETURN
      END
```

APPENDIX D: FORTRAN VARIABLES

A	Cross-section area (L^2). TRIDAG: The lower codiagonal.
AA	The coefficient matrix.
ACLOCK	The 24-hr clock time (T).
ADN	An empirical coefficient relating to the reactivity and diffusivity of nitrate in the benthal boundary layer ($T^{-BK}L^{-3BK}$).
AG	An empirical coefficient in the general equation for K2 ($T^{E1-2}L^{E2-E1}$).
AKC	An empirical coefficient relating the reactivity and diffusivity of the free variable in the benthal boundary layer ($T^{BK-1}L^{-3BK}$).
AKN	An empirical coefficient relating the reactivity and diffusivity of the ammonia variable in the benthal boundary layer ($T^{BK-1}L^{-3BK}$).
AKNX	An empirical coefficient relating the absorptivity and diffusivity of ammonia in the benthal boundary layer ($T^{BK-1}L^{-3BK}$).
AK1	An empirical coefficient relating the reactivity and diffusivity of carbonaceous biochemical oxygen demand ($T^{BK-1}L^{-3BK}$).
ALABEL	The names of each of the eight modeled species.
ALGADK	The rate of algal decay ($MT^{-1}L^{-3}$).
ALGAEB	The benthal concentration of algae (ML^{-2}).
ALGRO	The algal growth rate coefficient ($ML^{-3}T^{-1}$).
ALGO	Lumped algal decay rate coefficient ($ML^{-2}T^{-1}$).
ALG1	Lumped algal growth rate coefficient ($ML^{-2}T^{-1}$).
APO4	An empirical coefficient relating the absorptivity and diffusivity of phosphate in the benthal boundary layer ($T^{BK-1}L^{-3BK}$).
AS	Cross-sectional area averaged over the characteristic line (L^2).
ATB	The rate coefficient for bottom heat loss.
ATS	The rate coefficient for surface heat loss.
AUNIT	The units of each of the eight modeled species.
AO	RIV1H: A(I), RIV1Q: Area at the previous time-step.
A1	RIV1H: A(I+1), RIV1Q: Cubic interpolation coefficient for concentrations.
A2	Cubic interpolation coefficient for concentration.
A3	Cubic interpolation coefficient for concentration.
A4	Cubic interpolation coefficient for concentration.
B	Channel top-width (L). TRIDAG: The main diagonal.

Note: Appendix D may not contain all of the FORTRAN variables.

BC	Boundary conditions.
BCD	Downstream boundary conditions.
BCU	Upstream boundary conditions.
BETA	Momentum correction factor. TRIDAG: Workspace array.
BK	An empirical coefficient reflecting the thickness of the benthal boundary layer.
BLABEL	Array of labels, packed for output.
BOUND	Boundary conditions storage array.
BTD	Boundary condition type downstream.
BTU	Boundary condition type upstream.
BUNIT	Array of units, packed for output.
BO	RIV1H: Provisional estimate of B. RIV1Q: B at the previous time-step (L).
B1	Cubic interpolation coefficient for spatial derivative; of the concentration.
B2	Cubic interpolation coefficient for spatial derivative of the concentration.
B3	Cubic interpolation coefficient for spatial derivative of the concentration.
B4	Cubic interpolation coefficient for spatial derivative of the concentration.
C	RIV1H: The vector C_i of the matrix solution procedure. RIV1Q: The concentrations of each of the eight modeled species (ML^{-3}). TRIDAG: The upper codiagonal.
CBDOL	The concentration of carbonaceous biochemical oxygen demand in the lateral inflow (ML^{-3}).
CCL	The difference between concentration in the lateral inflow and the stream (ML^{-3}).
CL	The concentrations of each of the eight modeled species in the lateral inflow (ML^{-3}).
CLABEL	The label assigned to the free variable.
CLOCK	The clock time in fraction of a day (T).
CN	Lumped scalar equivalent of CN1(I).
CN1	Modified Manning's coefficient.
COEF	Coefficient in the rating curve.
COL	The concentration of the free variable in the lateral inflow (ML^{-3}).

CONST	The name of the namelist.
CONVRT	The conversion factor for customary to SI units.
COSP	The cosine of the junction angle.
CP	The presence/absence of each of the eight modeled species.
CSINK	The sink term for the free variable ($ML^{-3}T^{-1}$).
CUNIT	The units of the free variable.
CO	The scalar equivalent of C(I)(ML^{-3}).
COO	The concentration just upstream of the junction node (ML^{-3}).
C1	Coefficient in the stage-area formula.
C2	Coefficient in the stage-area formula.
C3	Coefficient in the stage-area formula.
D	RIV1H: A/B. MAT5: Gauss elimination factor. RIV1Q: Tributary dilution ratio.
DADX	Spatial derivative of area (L).
DAMK	Empirical coefficient in the formula for reaeration through control structures.
DAMKO	Scalar equivalent of DAMK.
DARK	Logical flag for whether the current time-step is outside of the daylight hours.
DATE	The date of the program's execution.
DAWN	Time of sunrise (T).
DBCA	Derivative of the rating curve with respect to A .
DBCQ	Derivative of the rating curve with respect to Q .
DBDH	Derivative of B with respect to H .
DC	Spatial derivative of the concentration (M^3L^{-4}).
DCOO	Spatial derivative of the concentration just upstream of the junction node (M^3L^{-4}).
DD	Dilution ratio of the tributary's spatial derivative.
DE	Elevation head loss over a reach (L).
DEN	Lumped denominator.
DFA	Derivative of the residual of the continuity equation with respect to A(I).
DFA1	Derivative of the residual of the continuity equation with respect to A(I+1).
DFQ	Derivative of the residual of the continuity equation with respect to Q(I).

DFQ1	Derivative of the residual of the continuity equation with respect to Q(I+1).
DGA	Derivative of the residual of the momentum equation with respect to A(I).
DGA1	Derivative of the residual of the momentum equation with respect to A(I+1).
DGQ	Derivative of the residual of the momentum equation with respect to Q(I).
DGQ1	Derivative of the residual of the momentum equation with respect to Q(I+1).
DJ	The main storage array.
DK	The spatial derivative of the decay rates ($T^{-1}L^{-1}$).
DOL	Dissolved oxygen concentration in the lateral inflow (ML^{-3}).
DOSAT	Dissolved oxygen saturation value (ML^{-3}).
DOSAVE	Dissolved oxygen concentration just upstream of a control structure (ML^{-3}).
DOX	Provisional estimate of dissolved oxygen concentration (ML^{-3}).
DQLA	Spatial derivative of the lateral inflow per unit area ($T^{-1}L^{-1}$).
DSINK	Spatial derivative of the source/sink terms ($T^{-1}L^{-1}$).
DT	Time increment (T).
DUDX	Spatial derivative of velocity (T^{-1}).
DX	Reach length (L).
DX1	Reach length (L).
DO	A distance equivalent of the time increment, used for estimating a derivative at the boundary (L).
D1	RIV1H: THETA*DT(T), RIVIQ: DX(1) (L).
D2	DWPMN: 2*D1/DX1(I) (TL^{-1}), RIV1Q: DX*(2) (L).
E	Fraction of the reach length above the node at which the characteristic line originated.
EC	The complement of E; i.e., 1-E.
ELO	Water surface elevation (L).
ELAPSE	Total elapsed simulation time (T).
ELO	Water surface elevation at the previous time-step (L).
EXPO	Exponent in the rating curve formula.
E1	An empirical coefficient for velocity in the general equation for k2.

E2	An empirical coefficient for mean depth in the general equation for k2.
F	A workspace array.
FEEDS	FEEDS(I) is the identification number of the segment that segment I feeds into.
FLIP1	A logical flag which indicates that the first two columns of the coefficient matrix have been pivoted.
FLIP2	A logical flag which indicates that the last two columns of the coefficient matrix have been pivoted.
FLOW	A scalar assigned the literal value 'Q' for scanning the input deck.
FMT	The output format array.
FMT1	The format specification F14.1.
FMT2	The format specification F14.2.
GAMMA	TRIDAG: A workspace array.
GR	Acceleration due to gravity (LT^{-2}).
GO	A correction factor for the spatial derivative due to accelerating flow.
G2	GR/2 (LT^{-2}).
H	RIV1H: depth of flow above the channel bottom (L), RIV1Q: average depth of flow (L).
HYDRO	The output array for hydrodynamic data.
HYDRO1	The input array for hydrodynamic data at odd time-steps.
HYDRO2	The input array for hydrodynamic data at even time-steps.
HO	RIV1H: a provisional estimate of H (L), RIV1Q: H at the previous time-step (L).
H1	A provisional estimate of H (L).
I	An index (usually for nodes).
IA	An index for the coefficient matrix.
IBC	An index for boundary conditions.
IBCL	IBC(L).
ICYCLE	An index for sensitivity analysis.
ID	Segment identification number.
IDAM	Control structure type.
IDAMO	A scalar equivalent of IDAM.
IDAY	The day number of the simulation.
IDLL	ID(LL).

IDC	A scalar equivalent of ID.
IEHOUR	The hours component of the 24-hr clock time.
IEMIN	The minutes component of the 24-hr clock time.
II	An index.
IND1	The total number of nodes in the system.
IND2	RIV1H: The array space needed for AA, RIV1Q: The number of time intervals.
IND3	RIV1H: The array space needed for C and R; RIV1Q: The total possible number of boundary condition values.
IND4	The array space required for HYDRO, HYDRO1, and HYDRO2.
IND5	The array space required for each component of HYDRO1 and HYDRO2.
INIT	The array of initial conditions (ML^{-3}).
IPRINT	The print interval.
IR	An index for the array R.
ISURMX	The maximum light intensity at the water surface (MT^{-3}).
IT	Indexes tributaries.
ITIME	Time-step number.
ITO	ITO(L) is the index of the T-array at which segment L deposits its data.
IT1	IT1(L) is the index of the T-array at which the first tributary to segment L deposits its data.
IT2	IT2(L) is the index of the T-array at which the last tributary to segment L deposits its data.
I3	The integer equivalent of C3.
J	An index (usually for time).
JBC	Segment L will find its boundary conditions at index JBC(L).
JBCD	Segment L will find its downstream boundary conditions at index JBCD(L).
JBCU	Segment L will find its upstream boundary conditions at index JBCU(L).
JJ	An index.
JNODE	Junction node.
JT	An array of junction nodes, packed in the same manner as T.
JUNCT	A flag for whether the current node is a junction node.
K	RIV1H: An index; RIV1Q: Decay rates for each of the eight modeled species (T^{-1}).

KALGDK The specific algal decay rate (T^{-1}).

KALGRC The specific algal growth rate ($T^{2}M^{-1}$).

KC The decay rate for the free variable (T^{-1}).

KDN The denitrification rate (T^{-1}).

KEXT The light extinction coefficient (L^{-1}).

KE1 Channel constriction energy loss coefficient.

KN The rate of ammonia decay due to nitrification (T^{-1}).

KNCBDN The Monod half-velocity constant for nitrate-inhibited denitrification (ML^{-3}).

KNX The rate of ammonia decay due to sediment sorption (T^{-1}).

KOALDK The Monod half-velocity constant for oxygen limitation of algal decay (ML^{-3}).

KOCBDN The Monod half-velocity constant for oxygen inhibition of denitrification (ML^{-3}).

KOCB1 The Monod half-velocity constant for oxygen limitation of carbonaceous biochemical oxygen demand decay (ML^{-3}).

KON The Monod half-velocity constant for oxygen limitation of denitrification (ML^{-3}).

KPO4DK The phosphate decay rate (T^{-1}).

KS The average K-rate across the characteristic line (T^{-1}).

KTB The bottom-water heat exchange rate (T^{-1}).

KTS The air-water heat exchange rate (T^{-1}).

K1 The CBOD decay rate (T^{-1}).

K2 The reaeration rate (T^{-1}).

L An index for segments.

LAMBDA The duration of daylight (T).

LAST The index of the last item interchanged.

LIB The set of boundary conditions for each segment.

LIMIT The index of the last item to be inspected.

LJ An index.

LL An index.

LO An index.

L1 An index.

M An index.

MAX The number of species actually modeled out of a total of eight possible.

MBC	Used in indexing boundary conditions.
MBOUND	The number of boundary conditions.
MC	The index of species.
MJ	An index.
MM	An index of labels and species.
MNODE	The number of nodes.
MP	The number of reaches.
MTIME	The number of time-steps.
MO	An index.
M1	The lower loop bound.
M2	The upper loop bound.
N	An index of species.
NBC	The number of boundary conditions at each time-step.
NDEPLO	The first node at which the nitrogen pool is depleted.
NDEPL1	The last node at which the nitrogen pool is depleted.
NH3NL	The concentration of ammonia in the lateral inflow (ML^{-3}).
NODE1	The index of the first node in each segment.
NOPO40	The first node at which the phosphate is depleted.
NOPO41	The last node at which the phosphate is depleted.
NO3NL	The concentration of nitrate in the lateral inflow (ML^{-3}).
NO3NX	A provisional estimate of the nitrate concentration (ML^{-3}).
NPOOL	The total inorganic nitrogen concentration--the sum of nitrate and ammonia (ML^{-3}).
NS	The number of segments in the system.
N1 through NS38	Locations within the main storage array.
ORDER	A collection of segment numbers in an upstream order.
ORGANL	The concentration of organic nitrogen in the lateral inflow.
P	The fraction of the nitrogen pool composed of nitrate.
PARM	A collection of parameters.
PI	π, 3.14159.
PO4L	The concentration of phosphate in the lateral inflow (ML^{-3}).
PO4X	A provisional estimate of phosphate concentration (ML^{-3}).
PO	A lumped friction loss term for node I($L^{-1/3}/T^{-1}$).
P1	A lumped friction loss term for node I+1($L^{-1/3}/T^{-1}$).
Q	The stream flow ($L^{3}T^{-1}$).

QL	Lateral inflow per unit of stream length (L^2T^{-1}).
QLA	Lateral inflow per unit of stream volume (T^{-1}).
QX	Q ($L^{3BK}T^{-BK}$).
QO	RIV1H: Q(I); RIV1Q: Q at the previous time-step.
Q1	Q(I+1).
R	RIV1H: The residuals from the governing equations; RIV1Q: The dissolved oxygen deficit ratio.
RCURVE	This is assigned a literal value of "R" and is used for scanning the input.
RMILE	River mile (L).
RMILEO	The river mile of the downstream terminus of the system (L).
RMSA	The root mean square of the initial cross-sectional areas at every node times the tolerance.
RMSQ	The root mean square of the initial flow at every node times the tolerance.
SAVE	A temporary storage location used in column pivoting.
SINI	The sine of the incident light angle.
SINK	The source/sink term for each of the eight modeled species ($ML^{-3}T^{-1}$).
SINKS	The average source/sink value across the characteristic line ($ML^{-3}T^{-1}$).
SNAME	The segment name.
SNAMEO	The segment name.
START	The start time of the simulation (T).
SUNSET	The time of sunset (T).
T	The tributary data transfer array.
TBIO	The temperature correction factor for biochemical processes.
TEMP	The temperature (°C).
TEMPL	The temperature of the lateral inflow (°C).
TEQ	The equilibrium temperature (°C).
THETA	The weighting factor.
TIME	The time of the program's execution.
TITLE	The title of the program run.
TNH3	The temperature correction factor for nitrification.
TOLER	The relative tolerance criterion for the Newton-Raphson procedure.
TPHYS	The temperature correction factor for physical processes.

TSINK	The source/sink term for temperature (°C T^{-1}).
TSIV	An empirical coefficient in the Tsivoglou-Wallace reaeration equation (L^{-1}).
ULL	The velocity at node I-1 of the previous time-step (LT^{-1}).
ULR	The velocity at node I of the previous time-step (LT^{-1}).
US	The average velocity across the characteristic line (LT^{-1}).
UUR	The velocity at node I of the current time-step (LT^{-1}).
V	TRIDAG: The solution vector.
XC	The previous time-step components of the continuity equation (L^2).
XM	The previous time-step components of the momentum equation (L^3T^{-1}).
Z	The channel bed elevation (L).