COMPUTER-BASED PROGNOSIS OF MISSOURI RIVER BED DEGRADATION
REFINEMENT OF COMPUTATIONAL PROCEDURES

by

F. M. Holly Jr., J. C. Yang and M. F. Karim

Sponsored by
Iowa State Water Resources Research Institute
355 Town Engineering Building
Ames, Iowa 50011

IIHR Report No. 281

Iowa Institute of Hydraulic Research
The University of Iowa
Iowa City, Iowa 52242
August 1984
# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Executive Summary</td>
<td>iii</td>
</tr>
<tr>
<td>Acknowledgements</td>
<td>v</td>
</tr>
<tr>
<td>List of Figures</td>
<td>vi</td>
</tr>
<tr>
<td>List of Tables</td>
<td>vii</td>
</tr>
<tr>
<td>I. Introduction</td>
<td>1</td>
</tr>
<tr>
<td>II. Review of IALLUVIAL Computer Program</td>
<td>3</td>
</tr>
<tr>
<td>III. Improvements in Computational Efficiency</td>
<td>15</td>
</tr>
<tr>
<td>IV. Iterative Coupling of Water Flow and Bed Evolution Equations</td>
<td>34</td>
</tr>
<tr>
<td>V. Boundary Conditions for Bed Elevation</td>
<td>38</td>
</tr>
<tr>
<td>VI. Recalculation of Missouri River Prognosis Simulation</td>
<td>51</td>
</tr>
<tr>
<td>VII. Conclusions and Suggested Further Research</td>
<td>59</td>
</tr>
<tr>
<td>References</td>
<td>60</td>
</tr>
<tr>
<td>Appendix A: IALLUVIAL Input Data Structure</td>
<td>61</td>
</tr>
<tr>
<td>Appendix B: IALLUVIAL Memory Structure</td>
<td>65</td>
</tr>
<tr>
<td>Appendix C: IALLUVIAL Source Listing</td>
<td>69</td>
</tr>
<tr>
<td>Appendix D: Sample Data Set</td>
<td>130</td>
</tr>
</tbody>
</table>
Executive Summary

Development of the Missouri River for navigation, flood control, and hydroelectric power production has been accompanied by a progressive lowering of its bed elevation between Yankton, South Dakota and Omaha, Nebraska. This bed lowering, or degradation, has caused numerous environmental and structural problems, including shrinking of oxbow lakes due to a lower water table, loss of wildlife habitat, potential undermining of bank protection works and bridge foundations, and reduced efficiency of water intake works.

Concern for the long-term effects of Missouri River bed degradation has led the Iowa State Water Resources Research Institute (ISWRRI) and the Omaha District, U.S. Army Corps of Engineers, to support the efforts of the Iowa Institute of Hydraulic Research in developing IALLUVIAL, a computer-based simulation model for predicting the future course of bed level changes. In a 1982-83 ISWRRI study, IALLUVIAL was used to develop prognosis of bed evolution over the next twenty years, for a variety of river management scenarios. The major conclusion of the study was that the worst of the degradation is probably over, only an additional three to four feet being forecast to occur over the eight feet already existing near Sioux City, Iowa. The study also suggested that the primary cause of the recent degradation is the navigation-dictated narrowing of the channel to 600 feet.

The present ISWRRI study had as its objective the investigation and solution of several conceptual and numerical problems in IALLUVIAL. These problems, which became apparent through the application of IALLUVIAL to an actual prototype situation, concerned excessive computation time; sensitivity to time and distance steps; singular behavior at the upstream model limit; and distortion of bed-perturbation propagation speed. This report describes various investigations aimed at understanding the above points, and presents several methodology and program changes designed to ameliorate them.

Computation time has been significantly reduced through adoption of a streamlined procedure for consulting Shields' diagram, and through use of a longer time step for backwater calculations than for sediment continuity calculations. It was concluded tentatively, on the basis of limited tests, that a backwater time step which is two to three times as large as the sediment continuity time step can be used without having a major effect on simulation accuracy.
A simultaneous solution of the water- and sediment-routing equations can be obtained through iterative coupling of the various processes involved. Such coupling has little effect for uniform bed sediments, but can reduce the computed degradation of beds affected by hydraulic sorting and armoring due to non-uniform sediments.

The bed-elevation singularity at the upstream limit of a model can be removed through the notion of a buffer zone within which disequilibrium between sediment inflow and transport capacity is tolerated. Use of this concept has greatly reduced the sensitivity of computed degradation at the upstream model limit to changes in time and distance steps.

Repetition of the base prognosis run reported in the 1982-83 ISWRRI study shows that neither the IALLUVIAL program modifications reported herein, nor the fundamental changes in armoring procedures recently introduced in a Corps of Engineers study, change the fundamental conclusion that the worst of the degradation is now over.

Future IALLUVIAL development should include improvement of the procedures for sediment routing by size fraction, and eventual incorporation of the IALLUVIAL methodology in a two-dimensional simulation model.
Acknowledgements

This study was performed under a grant from the Iowa State Water Resources Research Institute (ISWRRI), Office of Water Resources Technology Project No. C-001. Grateful acknowledgement is extended to ISWRRI for support of IALLUVIAL program development during both the 1982-83 and 1983-84 funding periods.

The IALLUVIAL computer program was originally developed with the support of the National Science Foundation and the Omaha District, U.S. Army Corps of Engineers. The Omaha District is continuing to support refinement and extension of certain methodologies in IALLUVIAL.

Dr. M. Fazle Karim, the developer of IALLUVIAL, participated in numerous technical discussions regarding the tests and program modifications performed under this study.
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>II.1</td>
<td>Subsurface Layers and Mixed Layer Composition</td>
<td>9</td>
</tr>
<tr>
<td>II.2</td>
<td>Schematic Representation of Tributary Inflows</td>
<td>11</td>
</tr>
<tr>
<td>II.3</td>
<td>Summary Block Diagram of IALLUVIAL</td>
<td>16</td>
</tr>
<tr>
<td>III.1</td>
<td>CPU Time Breakdown Before Program Modification</td>
<td>18</td>
</tr>
<tr>
<td>III.2</td>
<td>Discretization of Shield's Curve</td>
<td>19</td>
</tr>
<tr>
<td>III.3</td>
<td>Comparison of Delay Schemes for 30-day Update</td>
<td>25</td>
</tr>
<tr>
<td>III.4</td>
<td>Comparison of Delay Schemes for 60-day Update</td>
<td>26</td>
</tr>
<tr>
<td>III.5</td>
<td>Comparison of Delay Schemes for 120-day Update</td>
<td>27</td>
</tr>
<tr>
<td>III.6</td>
<td>Comparison of Three Update Intervals for Scheme B</td>
<td>28</td>
</tr>
<tr>
<td>III.7</td>
<td>Comparison of Three Update Intervals for Scheme C</td>
<td>29</td>
</tr>
<tr>
<td>III.8</td>
<td>Comparison of Three Update Intervals for Scheme D</td>
<td>30</td>
</tr>
<tr>
<td>III.9</td>
<td>Scheme D Performance after 20 Years</td>
<td>33</td>
</tr>
<tr>
<td>V.1</td>
<td>Sensitivity to Time Step for $\Delta x = 10$ miles</td>
<td>45</td>
</tr>
<tr>
<td>V.2</td>
<td>Sensitivity to Time Step for $\Delta x = 5$ miles</td>
<td>46</td>
</tr>
<tr>
<td>V.3</td>
<td>Sensitivity to Time Step for $\Delta x = 2$ miles</td>
<td>47</td>
</tr>
<tr>
<td>V.4</td>
<td>Sensitivity to Reach Length for $\Delta t = 30$ days</td>
<td>48</td>
</tr>
<tr>
<td>V.5</td>
<td>Sensitivity to Reach Length for $\Delta t = 15$ days</td>
<td>49</td>
</tr>
<tr>
<td>V.6</td>
<td>Sensitivity to Reach Length for $\Delta t = 5$ days</td>
<td>50</td>
</tr>
<tr>
<td>V.7</td>
<td>Demonstration of Apparent Instability at $\Delta x = 1$ mile, $\Delta t = 15$ days</td>
<td>52</td>
</tr>
<tr>
<td>V.8</td>
<td>Sensitivity to Assumed Degradation Wave Celerity at Upstream Boundary</td>
<td>53</td>
</tr>
<tr>
<td>V.9</td>
<td>Sensitivity to Iterative Coupling</td>
<td>54</td>
</tr>
<tr>
<td>VI.1</td>
<td>Comparison of Missouri River Prognosis Simulations Before and After IALLUVIAL Modification</td>
<td>56</td>
</tr>
</tbody>
</table>
VI.2 Comparison of Missouri River Prognosis Simulations for Various Computation Options (see text).................57

LIST OF TABLES

III.1 CPU Time Breakdown for Delayed Update Schemes.................23
III.2 Errors in Total Volume Removed after Ten Years for Various Delay Schemes..................................................32
V.1 Test Run Parameters..........................................................44
I. INTRODUCTION

Development of the Missouri River above Omaha for navigation, and its regulation for flood control and irrigation, have resulted in significant changes of river bed elevations. In particular, the bed has lowered as much as seven feet in places in the last 25 years, causing numerous potential and actual problems of bank protection stability, bridge foundation undermining, water intake structure efficiency, loss of wildlife habitat, loss of recreational sites, etc. On the other hand, the upstream regulation and channelization projects, executed by the U.S. Army Corps of Engineers, have admirably attained their objectives of creating a navigable waterway up to Sioux City, minimizing flood risk, and reclaiming riparian agricultural lands.

If bed lowering (referred to herein as degradation or scour) appears to be the penalty exacted by the river as the price to be paid for its beneficial development, then man must accommodate himself to this fact and learn to anticipate, if not to manage and control, the degradation. This straightforward goal is, however, much more easily stated than put into practice. Indeed, the problem of predicting bed evolution in alluvial rivers is an extremely difficult one, relying upon a current understanding of sediment transport processes in rivers which is, at best, extremely tentative and empirical. There are a great number of interdependent physical variables, and considerable uncertainty as to which are the independent and dependent ones. Available empirical sediment transport formulae often differ by orders of magnitude in their predictions for the same input data. Reliable field data are extremely difficult to obtain and interpret, while laboratory experiments can seldom simulate the complexity of natural situations.

The above difficulties notwithstanding, it has been possible to simulate the complex bed evolution process with some degree of confidence through the use of numerical techniques. Mathematical models incorporate detailed, deterministic, state-of-the-art understanding of the various physical processes involved, relying on the computer to synthesize them and solve the relevant conservation equations to simulate the overall bed evolution
process. In particular, the computer program IALLUVIAL has been developed specifically for prediction of bed evolution in the Missouri River (Karim and Kennedy, 1982; Karim and Holly, 1983; Holly and Karim, 1983; Karim, Holly, and Kennedy, 1984). Based on the success of its first use in an attempt to simulate the past twenty years of bed evolution between Gavins Point Dam and Omaha (Karim and Kennedy, 1982), IALLUVIAL was used to predict the next twenty years of evolution all along the Iowa border (Holly and Karim, 1983). That ISWRRI-sponsored study was beneficial for two major reasons. First, it showed that the worst of the degradation appears to be over, as only an additional three to four feet of degradation should occur before the year 2000. Second, it made it possible to identify several areas in the methodology of IALLUVIAL which need further attention. Two of these areas, namely a critical assessment of bed armoring procedures and sediment conservation by size fraction, have been and are currently under investigation through sponsorship by the U.S. Army Corps of Engineers, Omaha District. Two other areas, namely degradation time scale and computational efficiency, are the subject of the present study.

These latest improvements in IALLUVIAL, when added to its demonstrated success in simulating Missouri River bed evolution, have brought IALLUVIAL to the status of a mature, reliable predictor of Missouri River bed response to changes imposed on the river regime.

Although Chapter II of this report provides a brief summary of the principles and structure of IALLUVIAL, the reader totally unfamiliar with the program may wish to consult Karim and Kennedy (1982) for a general description of the technique and program before attempting to read Chapters III to V of the present report. These latter Chapters deal with the specific details of several computational procedures. Chapter VI describes the re-computation of the base Missouri River simulation from the 1982-83 ISWRRI study, using all the IALLUVIAL revisions and improvements which have been subsequently developed.
II. REVIEW OF IALLUVIAL COMPUTER PROGRAM

II.A. General Remarks. IALLUVIAL is a computer-based flow- and sediment-routing model for simulation of the long-term bed evolution of alluvial streams. It treats the flow as one-dimensional and quasi-steady, solving the following governing equations:

1. Equation of motion of water flow;
2. Equation of continuity for water flow;
3. A friction-factor equation that takes into account the variable roughness of sediment-transporting streams;
4. Equation for sediment discharge;
5. Equation of continuity for sediment.

A river reach is divided into subreaches, and the computations for each are performed for successive, discrete time intervals. In each time interval, IALLUVIAL solves the governing equations in two steps: the relations in 1 through 4 above are solved in the first "backwater" step to obtain water-surface elevation, depth, velocity, and sediment discharge at each computational point; in the second step, the sediment continuity equation is solved to yield depths of degradation/aggradation, changes in bed-material composition, and changes in armoring of the bed surface. The initial and boundary conditions required for a solution are: known initial bed elevation and sediment size distribution at all computation points; known water and sediment discharge hydrographs at the upstream limit of the model; and a known stage (water-surface elevation) hydrograph, or discharge-stage relationship, at the downstream limit of the model.

II.B. Sediment-discharge and friction-factor predictors. The Total-Load Transport Model (TLTM) developed by Karim and Kennedy (1981) is used for the sediment-discharge and friction-factor predictors. The formulation of TLTM takes into account the well-known fact that the friction factors of alluvial streams are heavily dependent on their sediment discharges, and avoids the need to specify a fixed hydraulic roughness, such as Manning's coefficient, a priori. In keeping with this concept, the friction-factor relation includes
sediment discharge as one of the independent variables, and an iteration scheme is used to calculate sediment discharge and friction factor from the following pair of simultaneous relations:

**Sediment-discharge predictor**

\[
\log \left( \frac{q_s}{\sqrt{g(s-1)D_{50}^3}} \right) = -2.2786 + 2.9719 \log V_1 + 1.0600 \log V_1 \log V_6 \\
+ 0.2989 \log V_2 \log V_6
\]  

(II.1)

**Friction-factor predictor**

\[
\log \left( \frac{U}{\sqrt{g(s-1)D_{50}}} \right) = 0.9045 + 0.1665 \log V_7 \\
+ 0.0831 \log V_4 \log V_5 \log V_7 + 0.2166 \log V_4 \log V_5 \\
- 0.0411 \log V_2 \log 3 \log V_4
\]  

(II.2)

where

\[
V_1 = \frac{U}{\sqrt{g(s-1)D_{50}}} , \quad V_2 = \frac{d}{D_{50}} , \quad V_3 = S \cdot 10^3 , \quad V_4 = \frac{u_*}{w} \\
V_5 = \frac{wD_{50}}{v} , \quad V_6 = \frac{u_* - u_c}{\sqrt{g(s-1)D_{50}}} , \quad V_7 = \frac{q_s}{\sqrt{g(s-1)D_{50}^3}}
\]

q_s = volumetric bed-material discharge/unit width
U = mean flow velocity
d = mean flow depth
D_{50} = median size of bed material
S = energy slope
w = fall velocity of sediment particles
v = kinematic viscosity of water
s = specific gravity of sediment particles
\[ u^* = \text{bed shear velocity} = \sqrt{gdS} \]
\[ u^*_{c} = \text{critical shear velocity obtained from Shields' diagram} \]

The numerical coefficients of Eqs. (II.1) and (II.2) were obtained through nonlinear regression analysis of extensive field and laboratory data (Karim and Kennedy, 1981).

**II.C. Water-surface-profile calculations.** Computations for sediment discharge and water surface profile in one time interval proceed simultaneously, because of the interdependence between friction factor and sediment discharge incorporated in Eqs. (II.1) and (II.2). Starting from a known or specified water-surface elevation at the downstream end, the calculation scheme simultaneously solves Eqs. (II.1), (II.2), and the steady-state continuity and energy equations of flow, by an iteration scheme analogous to the standard step method for backwater computations. This procedure calculates depth, velocity, energy slope, and sediment discharge at successive upstream sections in a single computational sweep, downstream to upstream.

**II.D. Change in bed elevation.** The depth of degradation or aggradation in a computational subreach of length \( \Delta x \) during a time interval \( \Delta t \) is calculated by applying the sediment-continuity equation between the two bounding computation points,

\[ (1-p) \frac{\partial z}{\partial t} + \frac{\partial d_S}{\partial x} = 0 \quad (\text{II.3}) \]

where \( p = \) porosity, and \( z = \) bed elevation. Equation (II.3) may be discretized to calculate the change in bed elevation for a reach, \( \Delta z \), from

\[ \Delta z = \frac{\Delta t}{\Delta x (1-p)} \left\{ \theta (q_{s_i}^{n+1} - q_{s_i+1}^{n+1}) + (1-\theta)(q_{s_i}^{n} - q_{s_i+1}^{n}) \right\} \quad (\text{II.4}) \]

in which \( q_{s_i} \) and \( q_{s_i+1} \) are sediment-transport capacities per unit width at downstream and upstream ends of the subreach respectively, \( n \) and \( n+1 \) denote successive times, and \( \theta \) is a weighting factor between 0 and 1. A positive value of \( \Delta z \) indicates degradation when \( (q_s)_i > (q_s)_{i+1} \), i.e., a deficit in sediment-transport capacities exists between the downstream and upstream
sections of the subreach. When \((q_S)_i < (q_S)_{i+1}\), \(\Delta z\) is negative, and its absolute value gives the depth of aggradation in the subreach. In the present version of IALLUVIAL, the entire wetted perimeter is shifted up or down by \(\Delta z\).

**II.E. Changes in bed-material composition.** The composition of an alluvial river bed undergoes continuous change in response to degradation or aggradation occurring due to imbalance in sediment transport capacities at the two ends of a reach. The depths (or volumes) of sediments of each size fraction scoured from or deposited in a reach are determined by applying the sediment continuity equation by size fraction. The fraction of sediment discharge in size interval \(k\) and reach \(i\), \(P_{di,k}\), is calculated from the relation given by Karim and Kennedy (1981):

\[
P_{di,k} = \frac{P_{i,k}D_{50i}^x}{\sum_{k=1}^{m} P_{i,k}D_{50i}^x}
\]

(II.5)

where \(P_{i,k}\) = fraction of size interval \(k\) in bed material of reach \(i\); \(D_{50i}\) = median bed-material size in subreach \(i\); \(D_k\) = geometric mean size of fraction \(k\); \(m\) = total number of sediment size intervals; and \(x\) is given by

\[
x = 0.0316 \left(\frac{d_i}{D_{50i}}\right)^{0.5}
\]

(II.6)

where \(d_i\) = average flow depth in subreach \(i\).

The size distribution of the bed sediments is updated at the end of each time interval by taking out the calculated depths of degradation from, or adding the deposited volumes to, the mixed layer, and then accounting for the proportionate change in each sediment size interval. The horizon of bed material immediately below the bed surface undergoing continual mixing due to agitation, overturning, bed-form migration, etc., is referred to herein as the mixed layer, and is assumed to have a thickness equal to the average bed-form (or dune) height and to be homogeneous in size distribution at any given time. The dune height \(H_d\), is estimated from the following relation (Allen, 1978):
$$H_d = d \left[ b_0 + b_1 \left( \frac{\phi}{3} \right) + b_2 \left( \frac{\phi}{3} \right)^2 + b_3 \left( \frac{\phi}{3} \right)^3 + b_4 \left( \frac{\phi}{3} \right)^4 \right]$$  \hspace{1cm} (II.7)

where $\phi$ = non-dimensional bed-shear stress; $b_0 = 0.079865$, $b_1 = 2.23897$, $b_2 = -18.1264$, $b_3 = 70.9001$, and $b_4 = -88.3293$. This relation accounts for both the growth and decay of bed-form heights with variation in bed-shear stresses.

**II.F. Bed armorimg.** In a degrading river, the finer sediment particles are transported preferentially from the bed, resulting in gradual coarsening of the bed surface. If the bed material contains sediments which are sufficiently large that they cannot be transported by the flow, then coarser particles gradually accumulate on the bed surface forming an "armor coat" which protects the underlying finer sediments which would otherwise be transported. The fraction of the bed surface, $A_f$, covered by these immobile sediment particles is expressed by the following relation, developed from volumetric considerations:

$$A_f(t) = C_1 (1-p) D_s(t) \frac{\sum_{k=1}^{m} P_k}{d}$$  \hspace{1cm} (II.8)

where $p$ = porosity; $d_s(t)$ = depth of degradation to time $t$; $P_k$ = fraction of bed material with size $D_k$; $\xi$ = sediment-size interval containing the smallest size which remains immobile on the bed; and $C_1$ = constant determined by the shape of the particles and their array on the bed. $C_1 = 1.90$ for ellipsoidal particles of shape factor $0.70$ laying flat in a one-diameter-thick armor layer.

Bed armorimg plays an important role in restoring balance between the sediment-transport capacity of the flow and the reduced sediment-supply rate into a reach. Armorimg assists the river in seeking a new equilibrium by reducing sediment discharge and also by changing the hydraulic roughness. It is assumed in IALLUVIAL that sediment discharge is reduced in direct proportion to the fraction of the bed surface that is armored ($A_f$). Similarly, the friction factor is taken equal to a weighted average of the fixed-bed roughness for the armored portion ($A_f$) and the movable bed roughness ($1 - A_f$). The thickness of the mixed layer is assumed to decrease linearly with increasing armorimg of the bed surface.
Although Eq. (II.8) is a valid general description of the armoring factor used in IALLUVIAL, research being conducted in parallel with the present study has led to use of a more refined procedure, whose details can be seen in the report by Karim, Holly, and Kennedy (1983).

II.G. Variation of bed material with depth. For a river degrading into its alluvium, the rate of degradation and its ultimate value are greatly influenced by the composition of underlying materials encountered by the degrading river bed. Underlying coarser materials result in increased coarsening of bed material and armoring of the bed surface, reducing both the time scale of evolution and the final depth of degradation; the reverse is true in the case of underlying finer materials. Variation in the composition of subsurface materials has been accounted for in IALLUVIAL by defining different layers of subsurface sediments (each layer defined by its thickness and sediment size distribution) below the initial bed elevation at the start of the simulation, as shown in Figure II.1. The thickness of each layer, \( t_L \), and size distribution of each layer \( P_i,k,L \) are defined as shown; other quantities shown in this figure are: \( d_s \) = depth of degradation, \( d_e \) = depth of materials entering mixed layer, \( (T_m)_t \) = mixed-layer thickness at time step \( t \), and \( (E_b)_t \) = bottom elevation of mixed layer at time \( t \). At the end of each time step during the simulation, the top and bottom elevations of the mixed layer are continuously tracked to determine their positions relative to the subsurface layers. The amounts of materials of different size distributions entering the mixed layer are computed accordingly, then are used to update the composition of the mixed-layer material. The amounts of coarser fractions which are responsible for armoring of the bed surface are also continuously updated. For example, referring to Figure II.1, the amount of material entering the mixed layer consist of a weighted average of three different layers of composition \( P_i,k,8\text{(partial)} \), \( P_i,k,9 \), and \( P_i,k,10\text{(partial)} \).

II.H. Sediment inputs from tributaries and bank erosion. The long-term evolution of alluvial river bed profiles is influenced by the tributaries in two ways: first, increased mainstream water discharge at the mouth of the tributary creates a backwater effect which alters the water surface elevations and sediment discharge capacities both upstream and downstream of the mouth; secondly, sediments brought in by the tributaries are added to the mainstem
Figure II.1. Subsurface Layers and Mixed Layer Composition.
sediment load delivered to the downstream reaches, often resulting in deposition of coarser particles near the tributary mouth in the short term. In the long term, however, water and sediment inflows from tributaries may lead to either degradation or aggradation in the neighboring reaches depending on the relative amounts of water and sediment discharges of the main river and the tributaries, and the relative size distribution of sediments transported by the main river and tributaries.

The influence of tributary inflows is considered in IALLUVIAL in two stages. First, the water surface profile is computed the same way as described in Section II.C, with water discharges at computation points adjusted to account for tributary water inflows. In the second stage, the application of the sediment continuity equation is modified as follows. Consider Figure II.2, where $Q_{i}$ and $Q_{S,i}$ = mainstem water and sediment discharges at node i; and $Q_{i}^t$, $Q_{S,i}^t$ = mainstem water and sediment discharges in the tributary entering at node i. With known flow depths, velocities, sediment discharges, and friction factors calculated in the first stage at the downstream side of each node, velocity and energy slope are recomputed at the upstream side of each node with mainstem discharge reduced by the amount of tributary inflow. For example, the modified velocity ($U_i'$) and energy slope ($S_i'$) at the upstream side of node i in Figure II.2 are calculated assuming the same water surface elevation (or depth, $d_i$) and friction factor ($f_i$) as at the downstream side:

$$U_i' = Q_{i+1}/A_i$$  \hspace{1cm} (II.9)

$$S_i' = \frac{f_i \cdot U_i'^2}{8g \cdot d_i}$$ \hspace{1cm} (II.10)

where $A_i$ = cross-sectional area corresponding to depth $d_i$, and $g$ = acceleration due to gravity. Sediment discharges are computed at points of tributary inflow from Eq. (II.1), e.g., at node i:

$$Q_{S,i}' = f(U_i', d_i, S_i', D_{50i})$$ \hspace{1cm} (II.11)
\[ Q_i = Q_{i+1} + Q_i^t \]
\[ Q_{i-1} = Q_i + Q_{i-1}^t \]

Figure II.2. Schematic Representation of Tributary Inflows.
where $Q_{s,i}'$ = recomputed values of $Q_{s,i}$, and $D_{50i}$ = median bed-material size at node $i$. The depths of degradation or aggradation for sediment fraction $k(\delta_{i,k})$ in the three reaches shown in Figure II.2 are obtained (for $Q_{s,i} > Q_{s,i+1}$) from:

$$\delta_{i,k} = \frac{(Q_{s,i}' - Q_{s,i+1}) \Delta t}{(1-p) B \Delta x} \cdot p_{di,k}$$

(II.12)

$$\delta_{i-1,k} = \frac{(Q_{s,i-1} - Q_{s,i}) \Delta t}{(1-p) B \Delta x} \cdot P_{di-1,k} - \frac{Q_{s,i}^t \Delta t}{(1-p) B \Delta x} \cdot P_{i,k}^t$$

(II.13)

$$\delta_{i-2,k} = \frac{(Q_{s,i-2} - Q_{s,i-1}) \Delta t}{(1-p) B \Delta x} \cdot P_{di-2,k}$$

$$- \frac{Q_{s,i-1}^t \Delta t}{(1-p) B \Delta x} \cdot P_{i-1,k}^t$$

(II.14)

where $\Delta t$ = time interval, $\Delta x$ = reach length, $p$ = porosity, $B$ = channel width, $P_{di,k}$ = fraction of sediment discharge in size interval $k$ and reach $i$ (Eq. (II.5)), and $P_{i,k}^t$ = fraction of sediment discharge in size interval $k$ for the tributary at node $i$. Note that the weighting parameter $\theta$ has been taken equal to 1 in these expressions. Positive values of $\delta_{i,k}$ in the above relations (II.12, II.13, and II.14) indicate net degradation, and negative values indicate net aggradation. The second term in Eqs. (II.13) and (II.14) gives the contribution from the tributary, and its magnitude relative to the first term (contribution from the main stream) determines the net amount of degradation or aggradation. A check on the consistency of sediment continuity (given by Eqs. (II.13) and (II.14)) is made for all fractions such that sediments brought in by tributaries are allowed to pass as suspended load (provided sufficient transport capacity is available at the downstream section), even though a negative value of $\delta_{i,k}$ is given by Eq. (II.13) or (II.14). This situation may arise when a finer-size fraction of bed material is removed completely in a previous time step, which makes $P_{di-i,k} = 0$ in Eq. (II.13), for example, and $\delta_{i-1,k}$ becomes negative (deposition) even though
the flow is capable of transporting sediments in that size interval. This procedure allows a partial check on the consistency of the sediment-continuity equation, a full treatment of which requires strict preservation of continuity of each sediment-size interval (contemplated in future refinement of IALLUVIAL). It may be noted that in the absence of tributaries, \( Q_{s,i}^{'} = Q_{s,i} \) for all \( i \) and the second term disappears in Eqs. (II.13) and (II.14). The above formulations take into account, though indirectly, the variations in the size distribution of sediment discharge between the main channel and tributaries.

For the cases when \( Q_{s,i} < Q_{s,i+1} \), Eqs. (II.12), (II.13), and (II.14) are modified by replacing \( p_{d,i,k} \)'s with \( p_{i,k} \)'s, where \( p_{i,k} \) = fraction of bed material in size interval \( k \) in reach \( i \).

Sediment inflows from tributaries are given as inputs to the program and are expressed as power-law functions of water discharge:

\[
Q_{s,i}^t = a_i (Q_i^t)^{b_i} \tag{II.15}
\]

where \( a_i \) and \( b_i \) are coefficients obtained from the sediment rating curve for the tributary at node \( i \) as determined by analysis of available data. \( Q_{s,i}^t \) = tributary sediment discharge (tons/day); and \( Q_i^t \) = tributary water discharge (cfs). Values of \( a_i \), \( b_i \) are entered as inputs to the program and are assumed to be constant in time. In considering sediment inflow at the upstream boundary, IALLUVIAL treats the upstream flow as one of the tributaries. Size distributions of tributary sediment inflows, \( p_{t,i,k} \), are given as inputs to the program.

Erosion of bank materials at high flows supplies a part of the sediment transport capacity of streamflow. This contribution is perhaps minor for natural river reaches, but may become significant in a reach downstream of a reservoir which traps virtually all incoming sediment load. The Missouri River reach between the Gavins Point Dam and Ponca (a length of about 57 miles) is essentially a meandering stream with erodible bankline and experiences considerable loss of land due to bank erosion (about 200 acres/year). The computer code of IALLUVIAL incorporates the effects of bank erosion on the evolution of the river bed.
The mechanics of bank erosion are complex and involve many controlling factors. Some important factors are stream discharge and water surface elevation and their rates of variation with time, composition of bank materials, channel location and alignment, wave heights, and groundwater elevation relative to the stream water surface elevation and rate of seepage. Quantification of the entire process, even if possible, would involve going beyond the one-dimensional framework of IALLUVIAL. Consequently the present formulation in IALLUVIAL adopts the following simple approach:

\[ Q_{i,k}^b = E_{bi} p_{i,k}^b; Q_i > Q_{\text{min}} \]  

(II.16)

\[ Q_{i,k}^b = 0; Q_i < Q_{\text{min}} \]  

(II.17)

where \( Q_{i,k}^b \) = rate of bank erosion in reach i for sediment size interval k; \( Q_i \) = water discharge in reach i; \( Q_{\text{min}} \) = minimum water discharge (cfs) above which erosion occurs; \( E_{bi} \) = user-specified bank erosion rate in reach i (cu.ft/mile/day); and \( p_{i,k}^b \) = fraction of bank-eroded material in reach i for size interval k. The values of \( E_{bi} \), \( Q_{\text{min}} \) and \( p_{i,k}^b \) are given as inputs to the program. An option to take \( p_{i,k}^b \) as equal to that of the bed-material size distribution has also been provided in the program. The sediment continuity equations, given by Eqs. (II.12) through (II.14), are modified due to bank erosion as follows:

\[ \delta_{i,k} = \delta_{i,k} \text{(by Eqs. II.12, II.13, or II.14)} - E_{bi} p_{i,k}^b \frac{\Delta t}{(1-p)^B} \]  

(II.18)

A check is made on sediment continuity, Eq. (II.18), as described earlier to assure that the eroded bank material of each size interval k is not deposited when the flow is capable of transporting it.

**III. Program organization.** The computer code of IALLUVIAL has been reorganized and considerably extended to incorporate additional features. Dynamic storage allocation of a large number of dimensioned arrays has been incorporated in the present code to optimize memory requirements and thus to enhance operational ease. The new version of IALLUVIAL consists of MAIN and 20 subroutines: INFLOW, SMAIN, SEDBED, START, CHANGE, DREDGE, WATRO, RESIS1,
TRASF, SECPRO, SLOAD, ARMOR, HYSORT, VSORT, SHIELD, TRIB, ERROR1, OUTPUT, QSUSBC, and RESFUN. An abbreviated block diagram of the program is shown in Figure II.3, which describes briefly the function of each subroutine and the flow of information among them.

Appendix A describes the input data structure of IALLUVIAL, and Appendix B presents the scheme used for dynamic allocation of memory. Appendix C is a complete listing of the FORTRAN IV source program as modified in this study. Appendix D presents a sample data set.

III. IMPROVEMENTS IN COMPUTATIONAL EFFICIENCY

III.A Justification. During the early development of IALLUVIAL, emphasis was necessarily placed on incorporation of the best possible methodologies for correct simulation of the relevant physical phenomena. The imperative of obtaining a reliable simulation far overshadowed concern for efficient use of computer resources, and rightly so. As a result, IALLUVIAL was, in its original form, quite demanding of computer memory and time. The relatively high cost per run limited the number of tests which could be performed, and the memory requirements limited the size of the model and the length of the simulation period.

In the previous ISWRRI study (Holly and Karim, 1983), extensive restructuring of program memory was performed to reduce memory requirements dramatically and automatically to tailor memory needs to the just the size needed for a particular run. This effectively removed any limitations on model size and simulation period as far as memory is concerned.

Included in the objectives of this continuation ISWRRI study was an analysis and reduction of computer time requirements. The first problem was to identify the time-consuming operations in IALLUVIAL so that a strategy for overall time reduction could be formulated. Accordingly, an internal CPU (Central Processing Unit) timer was implemented in the program to time various operations. The IBM assembler routine, called STEMPS, was adapted from one developed by SOGREAH Consulting Engineers, Grenoble, France. Within each subroutine, the elapsed time between each call and return was accumulated in a special variable for later output.
Figure II.3. Summary Block Diagram of IALLUVIAL.
The basic run used for CPU time analysis was a one-year simulation of the Missouri River model, using 30 global iterations in each 30-day time step (see Chapter IV below). In terms of computational effort, this is equivalent to a 30-year run with no iterations. Figure III.1 shows the resulting breakdown of CPU time. It should be noted that since the CPU time accumulated for a given subroutine includes the CPU time spent by lower-level subroutines which it calls, only the subroutines which manage water routing (WATPRO) and sediment routing (SLOAD) are mutually exclusive. Their combined CPU time is 99.4% of the total; the remaining 0.6% is spent in input/output operations and minor boundary data routines.

Figure III.1 is quite revealing as to the possibilities for CPU time reduction. First of all, the subroutine SHIELD, which does nothing more than consult Shield's curve and return a dimensionless shear stress upon demand, consumes nearly 17% of the total CPU time. Secondly, well over half the CPU time, nearly 63%, is consumed in operations related to water flow routing (backwater calculation). Since considerable effort had already been spent optimizing the arithmetic operations in all computational routines, it seemed apparent that two specific actions should be undertaken:

1) Investigate the seemingly high time requirements in SHIELD;

2) Investigate ways to perform the backwater calculation less often. These investigations and results are reported in the remainder of this chapter.

**III.B Economization of Shield's curve consultation.** Figure III.2 shows Shield's curve. From a purely computational point of view, the subroutine SHIELD is called with the Reynolds number ($R_*$) on the abscissa as an argument, and is required to "consult" the curve, then return the corresponding dimensionless critical shear stress ($\tau_*$) on the ordinate.

Since Shield's curve represents empirical observations, having no underlying simple mathematical structure, it is discretized as a series of six power functions for use in IALLUVIAL. Each segment is taken as

$$\tau_* = a R_*^b$$  \hspace{1cm} (III.1)
TOTAL CPU TIME = 74.84 SEC. FOR TEST RUN

ENCOMPASSES ALL WATER FLOW OPERATIONS

ENCOMPASSES ALL SEDIMENT ROUTING OPERATIONS

*SECPRO AND SHIELD APPEAR IN BOTH WATER FLOW AND SEDIMENT CONTINUITY CPU

Figure III.1. CPU Time Breakdown Before Program Modification.
Figure III.2. Discretization of Shield's Curve.

\[ \tau_\star = a R_\star^b \]
in which a and b, resulting from best-fit local approximations to the curve, are shown on Figure III.2. In the original version of the SHIELD routine, \( \tau_* \) was simply computed from Eq. (III.1) once the proper interval containing \( R_* \) was identified. However non-integer exponentiation is quite a costly function on digital computers, involving the equivalent of the order of 50 multiplications. Moreover, analysis of IALLUVIAL showed that SHIELD was called literally millions of times in a given run. This combination of a time-consuming unit operation and an extremely high frequency of execution explains the disproportionately large share of total time consumed by SHIELD as seen in Fig. III.1.

The solution to this problem involved a simple exchange of memory occupation for computer time. In an initial operation, values of \( \tau_* \) are computed for some 550 values of \( R_* \), spaced at 0.1 \( R_* \) units for \( 1 < R_* < 5 \), and at 1 \( R_* \) unit for \( 5 < R_* < 500 \). Thus Eq. (III.1) is executed only 550 times; the subsequent multitudinous calls to SHIELD use a simple, rapid addressing routine based on \( R_* \), to recover the appropriate discrete value of \( \tau_* \) from memory. The details can be seen in the program listing, Appendix C.

When the basic timing run was repeated with the new version of SHIELD, the CPU time consumed fell to that required for consultation of the timer. In other words, the new version consumes a negligible amount of time, for a 17% savings in total CPU time. Of course some new error may have been introduced by using discrete point values of \( \tau_* \) rather than interpolated ones. But one's intuitive feeling that this error would be negligible is confirmed by comparison of identical runs using the old and new versions of SHIELD; no differences in computed dependent variables can be detected.

**III.C. General strategy for reducing computer time consumed by water routing.** The computation procedure in WATPRO, presented in Karim and Kennedy (1982), carries out the water surface profile, sediment discharge, capacity and friction factor computation by using the methodology developed by Karim and Kennedy (1981). Most of the CPU time consumed in this calculation scheme is due to the iterative evaluation of Eqs. (II.1) and (II.2), which contain several logarithmic functions; these functions are quite time-consuming.
Normally IALLUVIAL computes the water surface profile, then computes bed elevation changes, in each time increment. Development of a technique for reducing CPU time for WATPRO is based on a major assumption of a weak relationship between water surface changes and the degradation/aggradation process within a short time period. This assumption, whose theoretical basis is discussed in Chapter 7 of Cunge et al. (1980), leads to the possible exploitation of a so-called "delayed updating" technique. The idea is that during a simulation run, the three primary variables friction factor, sediment discharge, and water surface elevation, will be updated only after a few time steps rather than every time step. The feasibility of this scheme can be evaluated by considering the degradation/aggradation deviations from a normal run without delayed update in the context of the reduced quantity of CPU time.

Three delayed updating schemes have been considered, as follows:

1) Complete delay of friction factor, sediment discharge, and water surface profile from bed elevation changes. For this case, WATPRO is deactivated completely for several time steps. In other words, the friction factor (F), sediment discharge (QS) and water surface profile (WS) are updated at some time interval which is greater than the basic time step used for updating the bed elevations, sorting, and armoring. Several delay intervals were used for the test runs to identify the optimum situation. The base time step used for the base test run is 15 days, and the simulation period is 10 years. The delay intervals which have been tested are 30, 45, 60, 75, 90 and 120 days. In the end, only the delay intervals of 30, 60, 120 days were retained for use in the other two delay schemes to be discussed below.

In the following sections, this technique will be represented by "D" and expressed as F-QS-WS constant for a certain time interval. In other words, the sediment continuity, armoring, and sorting computations are performed using the same values of F, QS, and WS for several basic time steps. The nomenclature D01 represents F-QS-WS held constant for 30 days; D02 represents F-QS-WS held constant for 60 days; D03 represents F-QS-WS held constant for 120 days; whereas the basic time step is 15 days.
2) Delaying friction factor and sediment discharge from water surface elevation and bed elevation changes. In this scheme, both the water surface profile and sediment computations are performed in each basic time step, but the friction factor and sediment discharge computed in RESIS1 are held constant for several time intervals, thus obviating the iterative evaluation of TLTM. This technique will be denoted as "C"; CO1 represents F-QS held constant for 30 days; CO2 represents F-QS held constant for 60 days; CO3 represents F-QS held constant for 120 days.

3) Delayed update of friction factor only. In this scheme, the water surface profile, sediment discharge, and sediment continuity operations are all performed in each basic time step, but with the friction factor computed in RESIS1 held constant for several time intervals. This still requires evaluation of Eq. (II.1), but obviates the need to obtain a simultaneous (iterative) solution of Eqs. (II.1) and (II.2). The symbol "B" is used to represent this scheme; B01 represents F held constant for 30 days; B02 represents F held constant for 60 days; B03 represents F held constant for 120 days.

III.D. Analysis of test runs. Analysis of the test runs involves weighing the saving in CPU time obtained by delayed update against the deterioration in simulation accuracy thus introduced. All comparisons are with respect to the base run A15, the presumption being that it represents a "true" simulation within the limits of IALLUVIAL's accuracy.

Table III.1 shows the breakdown of CPU time for the base run and the various delay test runs; only those subroutines consuming a non-negligible amount of time are shown. As expected, the routines concerned exclusively with sediment routing - SLOAD, ARMOR, HYSORT - as well as the general utility routines INFLOW and TRIB, show little or no variation among the various test cases. On the other hand, the CPU times in the water-routing-related routines - WATPRO, RESIS1, RESFUN, SECPR, TRASF - reflect the structure of the various schemes. Since the CPU time attributed to WATPRO incorporates all water routing operations, it is no surprise that its CPU time shows the scheme D - involving the most complete delay - to effect the greatest savings. As for the individual subroutines, the reader may find it interesting to note how their time varies as a function of the decoupling schemes used. It should be
### Table III.1. CPU Time Breakdown for Delayed Update Schemes

<table>
<thead>
<tr>
<th>Routine</th>
<th>Base Run</th>
<th>F delayed</th>
<th>F-QS delayed</th>
<th>F-QS-WS delayed</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A15</td>
<td>B01</td>
<td>B02</td>
<td>B03</td>
</tr>
<tr>
<td>ARMOR</td>
<td>2.61</td>
<td>2.62</td>
<td>2.64</td>
<td>2.63</td>
</tr>
<tr>
<td>HYSORT</td>
<td>3.15</td>
<td>3.14</td>
<td>3.15</td>
<td>3.14</td>
</tr>
<tr>
<td>INFLOW</td>
<td>0.12</td>
<td>0.12</td>
<td>0.12</td>
<td>0.12</td>
</tr>
<tr>
<td>RESFUN</td>
<td>12.74</td>
<td>7.03</td>
<td>3.86</td>
<td>2.19</td>
</tr>
<tr>
<td>RESIS</td>
<td>22.27</td>
<td>14.2</td>
<td>8.95</td>
<td>6.22</td>
</tr>
<tr>
<td>SECPRO</td>
<td>2.04</td>
<td>1.71</td>
<td>1.54</td>
<td>1.49</td>
</tr>
<tr>
<td>TRASF</td>
<td>6.77</td>
<td>6.81</td>
<td>6.83</td>
<td>6.80</td>
</tr>
<tr>
<td>TRIB</td>
<td>0.08</td>
<td>0.08</td>
<td>0.08</td>
<td>0.08</td>
</tr>
<tr>
<td>WATPRO</td>
<td>35.43</td>
<td>27.82</td>
<td>21.74</td>
<td>18.57</td>
</tr>
<tr>
<td>SMAIN</td>
<td>57.84</td>
<td>50.34</td>
<td>44.37</td>
<td>41.11</td>
</tr>
<tr>
<td>TOTAL</td>
<td>57.85</td>
<td>50.35</td>
<td>44.38</td>
<td>41.13</td>
</tr>
</tbody>
</table>
noted that the times for all routines incorporate a fixed time consumed by the call, timing function, and return, even if no actual computation is performed.

Analysis of the deterioration in accuracy for the various runs relies on qualitative comparison of the computed longitudinal bed profiles at ten years, the assumption being that the profile is the best overall indicator of simulation sensitivity to delayed update. Two specific questions must be answered:

1) Which of the three schemes causes the least deterioration of accuracy, for a given delay interval?

2) For a given scheme, how large can the delay interval be while maintaining the quality of the simulation?

Figures III.3-III.5 compare the longitudinal bed profiles computed with each delay scheme for delay intervals of 30, 60, and 90 days (base run time step 15 days). Close study of the figures shows that while all three schemes distort the bed profile, scheme D shows the least deterioration compared to run A15 - although it still underestimated the degradation by roughly a foot near RM 660. If this conclusion seems apparent for delay intervals of 30 and 60 days, it is not so for 120 days, where all three schemes badly distort the base degradation curve.

The apparent superiority of scheme D over schemes B and C is somewhat unexpected. One might have thought that the most complete delay would introduce the most error, but quite the opposite appears to be true for this particular test case. The explanation probably lies in the fact that the two-equation system of TLTM, Eqs. (II.1) and (II.2), is in a sense a finely tuned relation among sediment discharge capacity, friction factor, and water depth. When one of these variables is arbitrarily held constant, as in a delay scheme, then the system is thrown out of balance. In other words, complete delay appears preferable to partial delay.

Figures III.6-III.8 show the sensitivity of each scheme to the delay interval. It is clear that for all three schemes, an interval of 120 days (8 bed evolution time steps) is too long. For schemes B and C, the 30-day interval appears to give a profile slightly closer to the base run than the 60-day interval. For scheme D, no significant difference between the 30- and 60-day intervals is apparent.
Figure III.3. Comparison of Delay Schemes for 30-day Update.
Figure III.4. Comparison of Delay Schemes for 60-day Update.
Figure III.5. Comparison of Delay Schemes for 120-day Update.
Figure III.6. Comparison of Three Update Intervals for Scheme B.
Figure III.7. Comparison of Three Update Intervals for Scheme C.
Figure III.8. Comparison of Three Update Intervals for Scheme D.
These qualitative observations can be quantified by comparing the total volume of bed material removed in ten years for the different cases. Table III.2 presents the errors in volume compared to the base run A15. It is clear that for all three schemes, the 120-day delay interval introduces a total volume error of from 8% to 20%, which is unacceptably large. The -5.3% volume error for D02 is twice the error of B02 and C02, which somewhat belies the visual impression of D02's better fit to the base run profile, Fig. III.4. The 30-day interval volume error is largest for run C01 at -4.4%.

Although these figures do seem to disqualify the 120-day delay interval, they do not justify any further judgement as to the relative accuracy of the various schemes. This judgement is best made by visual estimation of best fit, tantamount to a least-squares comparisons as opposed to the pure arithmetic volume comparison.

From the above discussions, it would appear that the optimum delay scheme would be D02, i.e., complete delay at a 60-day (four bed-evolution time step) interval. An additional verification of this conclusion was effected by running schemes D01 and D02 for a 20-year simulation, and comparing the computed bed profiles with that of run A15 at 20 years. Figure III.9 shows the resulting comparison, from which it can be seen that the earlier conclusion of acceptability of a 60-day delay interval was perhaps too hastily made; there is in fact -12% of volume error at 20 years for the 60-day interval, but only +0.3% for 30-day interval.

These tests and conclusions can in no way be considered as complete or general. Once specific conclusion can however be drawn. For the Missouri River model with distance steps of about 10 miles and a time step of 15 days, the water surface elevation, sediment discharge capacity, friction factor, and sediment transport allocation by size fraction can be performed only every other time step without any significant deterioration of computed bed profiles. The CPU time saving thus effected is about 35%.

The potential advantage of this delayed update is for runs in which the time step must be very small for physical or numerical reasons, leading to expensive calculations. The savings to be effected by delaying the water routing computations by several base time steps may then become significant. A few tests of this kind are reported in Chapter VI below.
Table III.2  Errors in total volume removed after ten years for various delay schemes

<table>
<thead>
<tr>
<th>Run</th>
<th>A15</th>
<th>B01</th>
<th>B02</th>
<th>B03</th>
<th>C01</th>
<th>C02</th>
<th>C03</th>
<th>D01</th>
<th>D02</th>
<th>D03</th>
</tr>
</thead>
<tbody>
<tr>
<td>% error in computed volume removed</td>
<td>0</td>
<td>2.6</td>
<td>-2.6</td>
<td>-20.2</td>
<td>-4.4</td>
<td>+2.7</td>
<td>+7.9</td>
<td>-2.4</td>
<td>-5.3</td>
<td>-19.5</td>
</tr>
</tbody>
</table>

Volume removed in 10 years for run A15 = 1.392 x 10^6 ft³
Figure III.9. Scheme D Performance after 20 Years.
IV. ITERATIVE COUPLING OF WATER AND SEDIMENT ROUTING EQUATIONS

IV.A. Background and justification for uncoupled approach. The governing equations on which IALLUVIAL is based were referred to briefly in Section II.A of this report. It is useful for the present discussion to summarize these equations as follows:

Energy Equation for Steady Water Flow:

\[ \frac{d}{dx} (z + d + \frac{Q^2}{2gA^2}) = S_f \]  \hspace{1cm} (IV.1)

Sediment Discharge Predictor:

\[ F_1(Q_S, D_{50}, Q, A, d, S_f, ACF) = 0 \]  \hspace{1cm} (IV.2)

Friction Factor Predictor:

\[ F_2(Q, A, D_{50}, Q_S, S_f, d, ACF) = 0 \]  \hspace{1cm} (IV.3)

Sediment Continuity Equation:

\[ (1-p) B \frac{\partial z}{\partial t} + \frac{\partial Q_S}{\partial x} = 0 \]  \hspace{1cm} (IV.4)

Channel Geometry:

\[ A = A(d;x) \]  \hspace{1cm} (IV.5)

\[ B = B(d;x) \]  \hspace{1cm} (IV.6)

Hydraulic Sorting of Bed Material:

\[ D_{50}^n \rightarrow D_{50}^{n+1} \]  \hspace{1cm} (IV.7)
Armoring of Bed Surface:

\[ ACF_n + ACF_{n+1} \]  \hspace{1cm} (IV.8)

The eight dependent variables are:

- \( z \) = bed elevation;
- \( d \) = water depth;
- \( Q_s \) = sediment discharge;
- \( A \) = cross-sectional area;
- \( D_{50} \) = median bed material size;
- \( S_f \) = energy slope;
- \( B \) = water surface width;
- \( ACF \) = armoring factor;

The two independent variables are \( x \), the longitudinal coordinate; and \( t \), the time. The gravitational acceleration \( g \), the sediment porosity \( p \), and many other physical quantities which are independent of the bed evolution process appear in the functions \( F_1 \) and \( F_2 \). Eq. (IV.7) symbolizes the accounting operations which simulate the sorting, \( n \) and \( n+1 \) representing successive points in time. Equation (IV.8) symbolizes the additional accounting operations which simulate development of a stable armor layer.

The time-dependence of the bed evolution process, incorporated in the sediment continuity Eq. (IV.4), implies that all of the dependent variables are time-dependent. At any instant, the entire system of equations (IV.1) to (IV.6) must be simultaneously satisfied and consistent with the sorting and armoring processes of Eqs. (IV.7) and (IV.8). Were it possible to obtain an analytical solution to the entire system, this requirement of simultaneity would naturally be satisfied. But of course such an analytical solution cannot be obtained, due to the inherent nonlinearities; the tabular nature of Eqs. (IV.5) and (IV.6) for natural channels; the ad-hoc procedures (as opposed to mathematical relationships) of Eqs. (IV.7) and (IV.8); and the need to solve Eq. (IV.4) for each size fraction, followed by a reconstitution of the total change in bed elevation, \( z \).
Since no analytical solution is possible, IALLUVIAL uses a numerical method whose central feature is Preissmann's finite-difference approximation to the sediment continuity Eq. (IV.4):

\[
\frac{z_i^{n+1} - z_i^n + z_{i+1}^{n+1} - z_{i+1}^n}{\Delta t} = \frac{2\theta (Q_{Si}^{n+1} - Q_{Si}^n)}{(1-p)(B_i^{n+1} + B_{i+1}^{n+1})\Delta x} + \frac{2(1-\theta)(Q_{Si}^n - Q_{Si+1}^n)}{(1-p)(B_i^n + B_{i+1}^n)\Delta x} \tag{IV.9}
\]

in which the superscript \( n \) denotes the time level, the subscript \( i \) denotes the computational section, \( \Delta t \) is the computational time step, \( \Delta x \) is the distance between points \( i \) and \( i+1 \) (not necessarily constant), and \( \theta \) is a weighting factor between 0 and 1. Since all needed values are known at time level \( n \) (from the initial condition or from the results of the previous time step), the problem becomes one of solving the algebraic system of Eqs. (IV.1-IV.3), (IV.5-IV.9) in which the eight unknowns would be understood to be at time level \( n+1 \). But even this formal approach is not feasible, since Eqs. (IV.7) and (IV.8) are non-analytical accounting procedures which have not been expressed as closed-form mathematical expressions.

This apparent impossibility of obtaining a simultaneous solution to even the algebraic equivalent of the original differential system led to the adoption of a decoupling technique in the original version of IALLUVIAL. The solution proceeded in three stages:

1) Equations (IV.1-IV.3, IV.5, and IV.6) were solved in a "backwater sweep" from the downstream to the upstream boundary. During this sweep, the bed elevation \( z \), median diameter \( D_{50} \), and armoring factor \( ACF \) were held constant, as if the bed were temporarily frozen. The essential result of this sweep was the sediment transport capacity \( Q_{Si}^{n+1} \) at every point \( i \).

2) Equation (IV.9) was solved in a "downstream sweep" from the upstream to the downstream boundary to yield the new bed elevations \( z_i^{n+1} \) at each point \( i \). This sweep used \( \theta = 1.0 \), and treated the \( Q_{Si}^{n+1} \) values as constant, as if the sediment transport capacity were temporarily unaffected by bed elevation, armoring, or median sediment size changes.
3) The accounting processes of Eqs. (IV.7) and (IV.8) were finally executed using the degradation or aggradation computed in step (2) above.

This methodology is referred to as "uncoupled", since it assumes that the three processes occur sequentially, and not concurrently, within a given time step (of the order of several days). This apparent blatant violation of the principle of simultaneity of all mechanisms involved is rendered necessary by the practical difficulties associated with the lack of a closed-form representation of the armoring and sorting processes. Many numerical models of physical processes use such a decoupling procedure, whose validity rests on the assumption that the change in any one variable during a time step is small enough that its effect on the other variables (during the time step) can be ignored. In alluvial river models, the implication is that bed elevation changes in one time step are small compared to the depth of flow.

**IV.B. Need for a coupled approach.** Use of IALLUVIAL for simulation of certain idealized degradation and aggradation problems has brought to light some uncertainty about the dependence of the time scale of bed evolution on the computational grid parameters \( \Delta x \) and \( \Delta t \). Iterative coupling of the governing equations has been put forth as one possible solution to the time scale uncertainty. The objective is to obtain, through iterative repetition within one time step, a genuine simultaneous solution of Eqs. (IV.1-IV.3, IV.5-IV.9). It may seldom be necessary to seek such a simultaneous solution, the uncoupled approach being basically legitimate. But analysis of any numerical behavior, such as sensitivity to time and distance steps, can best proceed with the effects of uncoupling eliminated. Consequently the iterative coupling procedure described below has been implemented in IALLUVIAL.

**IV.C. Strategy for iterative coupling.** The iterative scheme adopted is straightforward in principle, though somewhat delicate to program. The scheme is best described by first noting that Eqs. (IV.7-IV.9) are the only ones which involve changes over time; the remaining equations of the set represent instantaneous consistency of the flow energy equation, TLTM, and channel geometry. With this fact in mind, one can readily appreciate the required sequence of iteration operations within one time step:
1) load imposed boundary conditions (mainstem and tributary water and sediment inflows, downstream water surface elevation).

2) using a fixed bed elevation z (latest estimate), and the latest estimates of D_{50} and ACF, compute the depth d, flow area A, energy slope S_f, water surface width B, and sediment discharge capacity Q_s at each computational point through simultaneous solution of Eqs. (IV.1-IV.3, IV.5-IV.6).

3) using the values of Q_s and B computed in step 2) above, compute the new estimates of bed surface elevation z^{n+1} from Eq. (IV.9).

4) using the change in bed surface elevations computed in (3) above, compute the new estimates of armoring factor ACF and median particle diameter D_{50} from Eqs. (IV.7) and (IV.8).

Steps (2) - (4) are repeated iteratively until successive estimates of z^{n+1} no longer change. When this convergence is reached, one is assured that the values of Q_s and B in Eq. (IV.9) result from simultaneous solution of Eqs. (IV.1-IV.3, IV.5-IV.8) at each time level n and n+1.

Tests of the above procedure are reported in Section VI below.

V. BOUNDARY CONDITIONS FOR BED ELEVATION

V.A. Formal boundary condition requirements. From a purely mathematical point of view, the boundary condition requirements for the quasi-steady water and sediment routing problem are clear. At each of N computation points there are four primary unknowns: water surface elevation y (or, equivalently, depth d), bed elevation z, sediment discharge capacity Q_s, and bed friction factor f (or, equivalently, energy slope S_f). The total number of unknowns is thus 4N.

At each of N points there is one sediment discharge predictor and one friction factor predictor. In each of (N-1) reaches there is one energy equation and one sediment continuity equation. Consequently there are 4N-2 equations for 4N unknowns, implying the need for two boundary conditions to close the system. There is no ambiguity as to what these two conditions should be; the backwater computation requires the imposition of a downstream
water surface elevation, and the sediment routing computation is meaningless without the imposition of an upstream sediment discharge entering the system.

**V.B Additional practical considerations.** If these formal requirements are clear, the practical ones are less so. The sediment continuity Eq. (II.4) computes $\Delta z$, the bed elevation change in the reach between two computational points. These reach elevations must be somehow allotted to computational points so that the subsequent backwater sweep can work with current thalweg elevations as the channel aggrades or degrades. In IALLUVIAL, this allocation consists of modifying the thalweg elevation at a computational point by the weighted average of the changes in the two adjacent computational reaches, i.e.

$$z_i^{n+1} - z_i^n = \frac{1}{\Delta x_{i-1} + \Delta x_i} (\Delta x_{i-1} \Delta z_{i-1} + \Delta x_i \Delta z_i) \quad (V.1)$$

in which it is understood that $\Delta z_i$ refers to reach $i$ and $z_i$ refers to point $i$. When the reaches are all of constant length $\Delta x$, Eq. (V.1) reduces to a simple averaging process.

The practical ambiguity arises from the fact that Eq. (V.1) cannot be written for points 1 (downstream) and N (upstream), as they have computational reaches on only one side. In the original version of IALLUVIAL, points 1 and N were simply attributed the bed elevation changes of reaches 1 and N-1, respectively. This practical expedient was not without its problems, however, as seen in the following two sections.

**V.C. Upstream boundary.** At the upstream boundary (point N), the difficulty is best appreciated by considering the sudden shut-off of upstream sediment inflow into a stream which was in equilibrium, i.e., transporting the imposed sediment load with no degradation or aggradation. In Eq.(II.4), if $i = N-1$, then in the first computational time interval after the shut-off of sediment inflow, $(Q_S)_{i+1} = 0$ while $(Q_S)_i$ remains the equilibrium sediment transport capacity. (Note $Q_S \equiv Bq_S$) Now whatever the values of $\Delta x$ and $\Delta t$, the degradation in the first time step will always be proportional to $\Delta t/\Delta x$, since $(Q_S)_N = 0$ and $(Q_S)_{N-1} = equilibrium$ value. For a given $\Delta x$, the degradation at point N is thus proportional to $\Delta t$, which is physically reasonable; as $\Delta t$ becomes smaller, the volume of material removed from the first reach

39
becomes smaller, and therefore the degradation in one time step is less. But for a given $\Delta t$, the degradation is inversely proportional to $\Delta x$; as $\Delta x$ becomes smaller, the volume of material removed from the first reach remains the same, but the degradation depth increases. In other words, the degradation thus computed in the first reach depends on $\Delta x$, and tends to infinity as $\Delta x$ tends to zero. This is a purely computational anomaly, having no physical basis, and thus warrants the investigation of an improved procedure as described below.

The key to eliminating the sensitivity of the bed elevation changes to $\Delta x$ at the upstream limit is adoption of a procedure described by Cunge (1980). The procedure makes no attempt to compute the bed level at the upstream point through the normal sediment continuity Eq. (II.4) or (IV.9). Instead, it implements the physical requirement that the channel bed level must ultimately change in such a way that its sediment transport capacity is equal to the imposed load. For example, if the imposed load is zero, then the channel must deepen until the transport capacity is zero; this becomes the mechanism for computing the bed level change at the upstream point.

This straightforward physical principle must nonetheless be slightly modified to account for the fact that the channel cannot instantaneously adjust to a change in the imposed load. Instead, it is assumed that some local degradation or aggradation due to imbalance between the imposed and transportable load can occur in a special computational reach $B\Delta x$ adjacent to the upstream limit. One seeks the bed elevation change which satisfies a special sediment continuity equation written for the "buffer" reach,

$$\{\phi (\bar{Q}_S^{n+1} - \bar{Q}_S^n) + (1-\phi)(\bar{Q}_S^n - \bar{Q}_S^n)\} = B\Delta x B_N (1-p) \Delta z_N$$ \hspace{1cm} (V.2)

in which $\bar{Q}_S$ = imposed sediment load at the upstream boundary, $\bar{Q}_S$ = TLTM sediment discharge capacity, $B_N$ = some appropriate width, $p$ = sediment porosity, and $\Delta z_N$ = change in bed elevation of reach $B\Delta x$ (and point N) in time $\Delta t$. In keeping with the usual uncoupled procedure (if only during one global iteration), the bed change $\Delta z_N$ is expressed as

$$\Delta z_N = z_{N}^{n+1} - z_{N}^{n} = y_{N}^{n+1} - d_{N}^{n+1} - z_{N}^{n}$$ \hspace{1cm} (V.3)
Here the water surface elevation \( y_N^{n+1} \) is known from the latest backwater sweep, and the previous bed elevation \( z_N^n \) is also known, leaving the depth \( d_N^{n+1} \) as an unknown in Eq. (V.3). Since \( Q_s^{n+1} \) and \( Q_s^n \) are given, and \( Q_s^n \) is known from the previous time step, the only remaining unknown is \( Q_s^{n+1} \), which can be thought of as the sediment transport capacity at the downstream end of reach \( B \Delta x \) with the armoring factor taken into account, i.e.

\[
Q_s^{n+1} = B_N(1-ACF_N) \tilde{q}_s^{n+1}.
\]

Again in keeping with the uncoupled context, one can consider \( \tilde{q}_s^{n+1} \) as a function only of \( q_N^{n+1} \) through the TLTM sediment discharge predictor, Eq. (II.1), all parameters other than \( d_N^{n+1} \) being known from the most recent backwater sweep and sorting/armoring operations. Consequently Eq. (V.2) reduces to a nonlinear algebraic equation in the single unknown \( q_N^{n+1} \), whose value can be determined through a Newton-Raphson iteration.

It is instructive to note that if one suppresses the buffer reach by setting \( \beta = 0 \), then the procedure outline above simply requires that the bed level adjust immediately so that the TLTM sediment discharge capacity at point \( N \) becomes equal to the imposed load. If \( \beta > 0 \), then the effect is to require the TLTM capacity to approach, but not equal, the imposed load, the difference being absorbed in aggradation or degradation in the buffer reach.

The value of \( B \Delta x \) is guided by the physical principle that the buffer reach should correspond roughly to the distance travelled by a bed perturbation in time \( \Delta t \). Denoting the bed perturbation celerity by \( c \), this yields

\[
\beta = c \Delta t / \Delta x
\]

The value of \( c \) is difficult to ascertain exactly, and depends on changing flow conditions and sediment composition. Current research at the Iowa Institute of Hydraulic Research (Karim and Kennedy, 1984) is directed toward developing estimators for \( c \). For the Missouri River, \( c \) would appear to be the order of 10 miles per year. However, the procedure does not appear to be particularly sensitive to \( \beta \), as is shown in the tests described in Section V.E. below.

Once the bed level at the upstream point has been determined as described above, a normal sediment continuity equation is applied to \( (1-\beta)\Delta x \) for use in
ultimately determining \( q_{n+1}^{N+1} \). In its present form, this equation uses the imposed load \( Q_s^{n+1} \) as inflow to the shortened reach, though an equally plausible argument could be made for using \( Q_s^{n+1} \). It is implicitly assumed - and virtually always true - that \( 0 < \beta < 1 \).

V.D. Downstream boundary. Practical difficulties at the downstream boundary are of a different nature. The problem arises because the sediment volume removed from the most downstream reach is obviously affected by \( Q_s^{n+1} \), the sediment transport capacity at the downstream limit of the model. Since \( Q_s^{n+1} \) depends strongly on the downstream depth \( d_1 \), any error in \( d_1 \) causes a concomitant error in \( Q_s^{n+1} \) which, in turn, causes an error in the volume of material removed from reach 1. Finally, when the equivalent bed elevation change in reach 1 is attributed to point 1, any error exacerbates the error in \( d_1 \) in subsequent time steps, and so on. The overall effect is that this accumulation of error may cause a bed disturbance to be initiated simply by the inexactness of the imposed downstream depth (or water surface elevation).

Whenever the downstream water surface elevation is imposed directly, as for flow into a lake, reservoir, or coastal area, the above error accumulation scenario should not occur. But in most riverine applications, no such definite condition exists, leaving no option but to impose some sort of rating curve \( d = d(Q) \). If such a relationship is not fully consistent with TLM (Eq. II.1 and II.2), then the resulting \( Q_s^{n+1} \) will contain error, inducing an artificial bed disturbance as described above. This is presently the case in IALLUVIAL; use of the downstream boundary condition option IDSWS = 1 provokes imposition of a depth calculated from a modified TLM equation in which \( q_s \) was eliminated as a variable. Consequently when the full TLM is applied to point 1, the resulting \( Q_s^{n+1} \) incorporates the effects of the approximate method, etc.

One solution to this anomaly would appear to be consistent use of the full TLM both for the rating curve and the backwater sweep; this would at least ensure that any uniform initial condition would remain so until physically disturbed, which is not presently the case. In particular, if one considers the bed elevation, friction factor, armoring factor, median particle diameter, etc., momentarily fixed (equal to their most recently computed values), then Eqs. (II.1) and (II.2) form two nonlinear algebraic equations in two unknowns, \( Q_s^{n+1} \) and \( y_1 \) (or, alternatively, \( d_1 \)). These can be solved by
Newton-Raphson iteration to yield $d_1$, with the assurance that this depth is consistent with the $q_S$ which will subsequently be computed in the backwater sweep (as long as the friction factor is changing slowly).

Development of IALLUVIAL subsequent to preparation of this report will test the above concept.

**V.E. Tests of sensitivity to time and distance steps.** The improvements discussed in Section IV and the present section were developed in response to deficiencies previously brought out by computation for both simplified channels and the Missouri River model. It does not appear necessary to present herein the evidence of the previous deficiencies. However, it is instructive to present the evidence of acceptable performance of the new procedures.

The test computations were performed for a prismatic, 1000-feet wide rectangular channel on a slope of one ft/mile and having uniform sediment size $D_{50} = 0.297$ mm. The downstream water surface was maintained at the level which assured a uniform flow depth of 8.65 feet at a discharge of $Q = 30,000$ cfs, corresponding to an equilibrium sediment concentration of 209,542 ppm. This initial uniform, equilibrium flow was perturbed by the complete shut-off of sediment inflow at the upstream boundary, initiating a degradation wave which propagated downstream. Table V.1 summarizes the parameters used for the various test runs, all of which adopted $\theta = 0.5$ and were terminated at 10 years' simulation, before the bed perturbation reached the downstream boundary.

**V.E.1. Sensitivity to $\Delta t$.** Figures V.1-V.3 show the computed bed profiles at ten years for time steps of 30,15, and 5 days, using distance steps of 10,5, and 2 miles, all with $\beta \Delta t/\Delta x = 12$ miles/year. Some systematic sensitivity to $\Delta t$ is seen at the upstream boundary, the smallest time step of 5 days showing as much as 10% more degradation than 30 days. However, the effect of $\Delta t$ appears negligible at all other computation points.

**V.E.2. Sensitivity to $\Delta x$.** Figures V.4-V.6 show the computed bed profiles for distance steps of 10, 5, and 2 miles, at time steps of 30, 15, and 5 days. The degradation is slightly but systematically greater for 10 miles, though the severe sensitivity at the upstream point is now noticeably
### Table V.1. Test Run Parameters

<table>
<thead>
<tr>
<th>RUN</th>
<th>( \Delta x ) miles</th>
<th>( \Delta t ) days</th>
<th>( \Delta x/\Delta t ) miles/yr</th>
<th>( \beta )</th>
<th>( \beta \Delta x/\Delta t ) miles/yr</th>
<th>Global Iterations</th>
<th>( \Delta t/\Delta x^2 ) days/mi²</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>30</td>
<td>120</td>
<td>0.1</td>
<td>12</td>
<td>no</td>
<td>0.3</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>15</td>
<td>240</td>
<td>0.05</td>
<td>12</td>
<td>no</td>
<td>0.15</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>5</td>
<td>720</td>
<td>0.0167</td>
<td>12</td>
<td>no</td>
<td>0.05</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>30</td>
<td>60</td>
<td>0.2</td>
<td>12</td>
<td>no</td>
<td>1.2</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>15</td>
<td>120</td>
<td>0.1</td>
<td>12</td>
<td>no</td>
<td>0.6</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>5</td>
<td>360</td>
<td>0.0333</td>
<td>12</td>
<td>no</td>
<td>0.2</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>15</td>
<td>24</td>
<td>0.5</td>
<td>12</td>
<td>no</td>
<td>15</td>
</tr>
<tr>
<td>8</td>
<td>2</td>
<td>30</td>
<td>24</td>
<td>0.5</td>
<td>12</td>
<td>no</td>
<td>7.5</td>
</tr>
<tr>
<td>9</td>
<td>2</td>
<td>15</td>
<td>48</td>
<td>0.25</td>
<td>12</td>
<td>no</td>
<td>3.75</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>5</td>
<td>144</td>
<td>0.0833</td>
<td>12</td>
<td>no</td>
<td>1.25</td>
</tr>
<tr>
<td>11</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>12</td>
<td>10</td>
<td>15</td>
<td>240</td>
<td>0.1</td>
<td>24</td>
<td>no</td>
<td>0.15</td>
</tr>
<tr>
<td>13</td>
<td>10</td>
<td>5</td>
<td>720</td>
<td>0.1</td>
<td>72</td>
<td>no</td>
<td>0.05</td>
</tr>
<tr>
<td>14</td>
<td>5</td>
<td>30</td>
<td>60</td>
<td>0.1</td>
<td>6</td>
<td>no</td>
<td>1.2</td>
</tr>
<tr>
<td>15</td>
<td>5</td>
<td>5</td>
<td>360</td>
<td>0.1</td>
<td>36</td>
<td>no</td>
<td>0.2</td>
</tr>
<tr>
<td>16</td>
<td>10</td>
<td>30</td>
<td>120</td>
<td>0.1</td>
<td>12</td>
<td>yes</td>
<td>0.3</td>
</tr>
</tbody>
</table>
Figure V.1. Sensitivity to Time Step for $\Delta x = 10$ miles.
Figure V.2. Sensitivity to Time Step for \( \Delta x = 5 \) miles.
Figure V.3. Sensitivity to Time Step for $\Delta x = 2$ miles.
Figure V.4. Sensitivity to Reach Length for $\Delta t = 30$ days.
Figure V.5. Sensitivity to Reach Length for $\Delta t = 15$ days.
Figure V.6. Sensitivity to Reach Length for $\Delta t = 5$ days.
absent, the maximum difference due to $\Delta x$ being less than 3% at the upstream point. Figure V.7 shows, however, that when $\Delta x$ is further reduced to 1 mile with $\Delta t = 15$ days, a dramatic change in bed profile appears. This is probably due to unstable behavior. Although no formal stability analysis has been performed, it is likely that the numerical methods employed in IALLUVIAL are subject to a diffusion-type stability parameter proportional to $\Delta t/\Delta x^2$. Comparison of values in Table V.1 suggests that for this particular computation, $\Delta t/\Delta x^2$ must be roughly 10 days/mi$^2$ or less for stability.

V.E.3 Sensitivity to Bed Wave Celerity. When $\phi$ is held constant in Eq. (V.4), variations in $\Delta x$ and $\Delta t$ effectively change the value of bed wave celerity $c$ adopted for the buffer reach length used for the upstream boundary. Figure V.8 shows the bed profiles calculated for a range of celerities for 6 to 72 miles/year. It is apparent that the sensitivity of the degradation at the upstream point to $c$ is no greater than to $\Delta x$ and $\Delta t$ as seen earlier. This is most encouraging, as it is generally difficult to estimate $c$ with any degree of reliability.

V.E.4. Effect of Global Iterations. It became apparent during tests of the global iteration scheme outlined in Section IV that for uniform sediment, the physical coupling between transport capacity and depth of degradation is so weak that iterations would have little practical benefit despite their theoretical attractiveness. Figure V.9 shows this to be the case; curves 1 and 4, representing the same parameters without and with iterations, are virtually identical. The effect of iterations is expected to be more pronounced when non-uniform sediments are used, as is shown in Section VI.

VI. RECALCULATION OF MISSOURI RIVER PROGNOSIS SIMULATION

Since the 1982-83 ISWRRI study (Holly and Karim, 1983) reached rather definite conclusions as to the future course of degradation in the Missouri River, it was considered important to verify that the IALLUVIAL methodology changes introduced since that study do not fundamentally change its conclusions.
Figure V.7. Demonstration of Apparent Instability at $\Delta x = 1$ mile, $\Delta t = 15$ days.
LONGITUDINAL PROFILES OF DEGRADATION AT 10 YRS

Figure V.8. Sensitivity to Assumed Degradation Wave Celerity at Upstream Boundary.

DEGR. (FT)

2.0 2.0 2.0 2.0 2.0 2.0

RUN 1: X1010T30 BETA = 1
RUN 2: X1010T15 BETA = 1
RUN 3: X1010T5S BETA = 1
RUN 4: X510T30 BETA = 1
RUN 5: X510T5S BETA = 1
RUN 6: X510T5S BETA = 1
Figure V.9. Sensitivity to Iterative Coupling.
The prognosis run chosen for the comparison is Run S1 reported by Holly and Karim (1983). This base run involves a simulation of the next 20 years of bed evolution from Gavins Point Dam to Iowa's Southern border under present conditions of upstream water release schedules, tributary hydrology, bank erosion, and channelization.

Figure VI.1 shows the computed longitudinal profiles of degradation after 20 years of simulation for the original Run S1 (curve 1) and Run NEW1 (curve 2), the latter using exactly the same model data set and hydrologic inputs, but incorporating the new armoring procedures described by Karim, Holly, and Kennedy (1983), the revised upstream bed level methodology described in Section V of this report, the global iteration scheme described in Section IV, the streamlined Shield's curve consultation described in Section III, and numerous minor modifications to the computational procedures.

The only significant difference in the two simulations is from RM 615 to RM 580, where the NEW1 calculation predicts significantly more aggradation than S1. It is difficult a priori to judge whether the S1 or NEW1 tendency is the more correct one; however, experience with simulation of the past twenty years (Karim and Kennedy, 1982) suggests that the more moderate aggradation of S1 is probably closer to reality. The excessive aggradation of NEW1 appears to be the result of some interaction between the global iteration scheme (see below) and the known uncertain behavior of IALLUVIAL's size-fraction-decomposition methodology under global aggradation conditions. This methodology is presently being reviewed with support from the Omaha District, U.S. Army Corps of Engineers.

Upstream of RM 620, NEW1 predicts systematically less degradation than S1, the average reduction being the order of 0.5 feet, with a maximum reduction of 1.5 ft near RM 760. The new upstream bed level methodology produces about a 0.5-foot reduction in degradation at Gavins Point Dam.

Additional insight into the effects of the various modifications can be gleaned from Fig. VI.2, which compares S1 and NEW1 with runs NEW2 and NEW3; NEW2 is the same as NEW1 but without global iterations, and NEW3 adopts a 5-day basic time step (as opposed to 15 days for the other runs) but with backwater updating at 15-day intervals (analogous to delay routine DO3 of
Figure VI.1. Comparison of Missouri River Prognosis Simulations Before and After IALLUVAL Modification.
Figure VI.2. Comparison of Missouri River Prognosis Simulations for Various Computation Options (see text).
Section III.B). The most noteworthy feature of Fig. VI.2 is that with the exception of run NEW1's excessive downstream aggradation, all four runs predict the same general overall bed evolution patterns. It is interesting to compare NEW1 coupling through (global iterations) and NEW2 (no coupling); NEW1 generally predicts less degradation than NEW2, reflecting the strong physical coupling between degradation and bed coarsening/armorling. This contrasts with the nearly identical degradation of runs 1 and 16 on Fig. V.9, for which the physical coupling was very weak, uniform sediments precluding any coarsening or armorling.

This contrast of behavior between the effect of iterations in the uniform and nonuniform cases can be thought of qualitatively as follows. In uniform sediments, the sediment discharge capacity $Q_s$ depends essentially (and only weakly) on the depth $d$. The small change in depth due to degradation in one time step has such a small effect on $Q_s$ that the iterative correction is insignificant. In nonuniform sediments, $Q_s$ depends quite strongly on the median bed layer particle size $D_{50}$ and the armorling factor $ACF$. The degradation in one time step changes $D_{50}$ and $ACF$ enough that an iterative correction has a significant effect. In pure degradation, the iteratively-coupled calculation will always effectively encounter a larger $D_{50}$ and $ACF$ than the uncoupled case, ultimately reducing the total degradation.

The overall conclusion to be drawn from Figs. VI.1 and VI.2 is that the improvements in IALLUVIAL do not change the overall conclusions of Holly, Karim, and Kennedy (1983) regarding the future course of Missouri River bed degradation above Omaha.

It is interesting to note that relative cost of the runs discussed above, in terms of CPU time:

- Run S1: 7 min 47 sec CPU time
- Run NEW1: 12 min 31 sec CPU time
- Run NEW2: 4 min 56 sec CPU time
- Run NEW3: 7 min 41 sec CPU time

The essential computational economy introduced in this study is seen in comparing S1 and NEW2. The time penalty of iterative coupling is seen in
comparing NEW1 and NEW2. The effect of delayed backwater update in minimizing the time penalty of using a smaller base time step is seen in comparing NEW2 and NEW3.

VII. CONCLUSIONS AND SUGGESTIONS FOR FURTHER RESEARCH

The improvements to IALLUVIAL implemented under this study have significantly increased its efficiency and reliability. One additional major improvement, reformulation of sediment routing by size fraction, is currently underway through sponsorship by the Omaha District, U.S. Army Corps of Engineers. Repetition of the Missouri River prognosis simulation previously reported shows that the program modifications reported herein do not change the fundamental prediction that the worst of the bed degradation is over.

Two remaining points of some importance should be investigated as soon as practicable. One is implementation and testing of the downstream bed elevation routine outlined in Section V.D. The other, not discussed in this report, is the development and testing of a Newton-Raphson iteration scheme for simultaneous solution of Eqs. (II.1), (II.2), and the flow energy (backwater) equation. The procedure presently used is somewhat ad hoc, inefficient, and occasionally subject to non-convergence. Given the large proportion of computation cost consumed by the backwater sweep, as documented in Section III.A, efforts should be made to profit from the Newton-Raphson speed and reliability of convergence.

In terms of long-term development, the time is fast approaching when the cumulative experience with TLTM and IALLUVIAL should be brought to bear on the problem of two-dimensional (depth-averaged) modeling in wide alluvial stream. The conceptual and numerical problems involved are legion; but to neglect research in this direction is to risk attempting to extract more information than is inherently legitimately available from a one-dimensional model.
REFERENCES


### APPENDIX A. ALLUVIAL INPUT DATA PROCEDURE

<table>
<thead>
<tr>
<th>Card No.</th>
<th>Variable Name</th>
<th>Format Column</th>
<th>Variable description and remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>*1</td>
<td>TITLE</td>
<td>15A4(1-60)</td>
<td>Title of the study or run</td>
</tr>
<tr>
<td>*2</td>
<td>N</td>
<td>I5(1-5)</td>
<td>Number of computational points or cross-sections</td>
</tr>
<tr>
<td></td>
<td>N1</td>
<td>I5(6-10)</td>
<td>Number of sediment size fractions</td>
</tr>
<tr>
<td></td>
<td>NT</td>
<td>I5(11-15)</td>
<td>Max. no. of time steps</td>
</tr>
<tr>
<td></td>
<td>MAXMA</td>
<td>I5(16-20)</td>
<td>Max. no. of elevations used to define cross-sections</td>
</tr>
<tr>
<td></td>
<td>NTRIB</td>
<td>I5(21-25)</td>
<td>Number of tributaries, NTRIB &gt; 1</td>
</tr>
<tr>
<td></td>
<td>NBANK</td>
<td>I5(26-30)</td>
<td>Number of reaches with erodible bank materials</td>
</tr>
<tr>
<td></td>
<td>IBED</td>
<td>I5(31-35)</td>
<td>0 if vertically homogeneous bed material; 1 otherwise</td>
</tr>
<tr>
<td></td>
<td>MAXBED1</td>
<td>I5(36-40)</td>
<td>Max. no. of nonhomogeneous vertical layers in a reach</td>
</tr>
<tr>
<td></td>
<td>IDREJ</td>
<td>I5(41-45)</td>
<td>0 if no dredging; 1 otherwise</td>
</tr>
<tr>
<td></td>
<td>NTY</td>
<td>I5(46-50)</td>
<td>Time step in days</td>
</tr>
<tr>
<td></td>
<td>IDCUP</td>
<td>I5(51-55)</td>
<td>No. of decoupling time steps (&gt; 1)</td>
</tr>
<tr>
<td>*3</td>
<td>INPR</td>
<td>I5(1-5)</td>
<td>Frequency of printed output; in days</td>
</tr>
<tr>
<td></td>
<td>IROCK</td>
<td>I5(6-10)</td>
<td>0 if no rock outcrops; 1 otherwise</td>
</tr>
<tr>
<td></td>
<td>ICHGDT</td>
<td>I5(11-15)</td>
<td>No. of changes in time step</td>
</tr>
<tr>
<td></td>
<td>UPS</td>
<td>I5(16-20)</td>
<td>0 for constant upstream sediment inflow; 1 for upstream sediment inflow rating curve</td>
</tr>
<tr>
<td></td>
<td>IBUG2</td>
<td>I5(21-25)</td>
<td>0 for normal output; 1 for extensive diagnostic messages</td>
</tr>
<tr>
<td></td>
<td>ICHB</td>
<td>I5(26-30)</td>
<td>0 for no width changes with time; 1 otherwise</td>
</tr>
<tr>
<td></td>
<td>ICOFF</td>
<td>I5(31-35)</td>
<td>0 for no cutoff with time; 1 otherwise</td>
</tr>
<tr>
<td></td>
<td>ITMAX</td>
<td>I5(36-40)</td>
<td>Maximum number of global iterations in each time step</td>
</tr>
<tr>
<td></td>
<td>NRES3</td>
<td>I5(41-45)</td>
<td>Results file reference number; 0 for no results file</td>
</tr>
<tr>
<td></td>
<td>LASTIM</td>
<td>I5(46-50)</td>
<td>Total number of days to be simulated</td>
</tr>
<tr>
<td></td>
<td>IDSWS4</td>
<td>I5(51-55)</td>
<td>0 if downstream W.S. elev. given as input; 1 otherwise</td>
</tr>
<tr>
<td></td>
<td>NRESTA</td>
<td>I5(56-60)</td>
<td>Reference no. for results file to be read; 0 no file</td>
</tr>
<tr>
<td></td>
<td>NEWSTA</td>
<td>I5(61-65)</td>
<td>Indicating the time of result file data to be read, in days</td>
</tr>
<tr>
<td>*4</td>
<td>ALFA</td>
<td>F10.4(1-10)</td>
<td>Parameter $C_2$ in Eq. (4) of Ref. (6)</td>
</tr>
<tr>
<td></td>
<td>BETA</td>
<td>F10.4(11-20)</td>
<td>Parameter $C_3$ in Eq. (8) of Ref. (6)</td>
</tr>
<tr>
<td></td>
<td>C21</td>
<td>F10.4(21-30)</td>
<td>Required value = 1.0</td>
</tr>
<tr>
<td></td>
<td>C22</td>
<td>F10.4(31-40)</td>
<td>Required value = 1.0</td>
</tr>
<tr>
<td></td>
<td>FMIX</td>
<td>F10.4(41-50)</td>
<td>Required value = 1.0</td>
</tr>
<tr>
<td></td>
<td>CARM5</td>
<td>F10.4(51-60)</td>
<td>Parameter $C_1$ in Eq. (3) of Ref. (6)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>*5</td>
<td>STR6</td>
<td>F14.7(1-14)</td>
<td>Energy slope for d.s. boundary and initial estimate</td>
</tr>
<tr>
<td></td>
<td>CRITER</td>
<td>F14.7(15-28)</td>
<td>Criterion for global iteration control</td>
</tr>
<tr>
<td></td>
<td>THETA</td>
<td>F14.7(29-42)</td>
<td>Time weighting parameter for sediment continuity</td>
</tr>
<tr>
<td>*67</td>
<td>DMZ(I)=1,N</td>
<td>6F10.4</td>
<td>Depth of rock outcrop below initial bed by point</td>
</tr>
<tr>
<td>7</td>
<td>LOCTR(7), I = 1 NTRIB</td>
<td>12I5</td>
<td>Locations (point numbers) of tributaries, upstream to downstream LOCTR(1) = N always</td>
</tr>
<tr>
<td>88</td>
<td>AC(L), BC(L), L = 1, NTRIB</td>
<td>2F14.8</td>
<td>Sediment rating curve coefficients, Eq. (II.13) one card for each tributary</td>
</tr>
<tr>
<td>*9</td>
<td>PTRIB (L,K) K = 1, N1, L = 1, NTRIB</td>
<td>6F10.2</td>
<td>Distribution function for tributary sediment loads; k varies first</td>
</tr>
<tr>
<td>109</td>
<td>IBSED</td>
<td>I5(1-5)</td>
<td>0 if eroded bank material same as bed; 1 otherwise</td>
</tr>
<tr>
<td></td>
<td>QMIN</td>
<td>F10.1(6-15)</td>
<td>Water discharge (cfs) below which no bank erosion occurs</td>
</tr>
<tr>
<td>119</td>
<td>LOCBER(M) M = 1, NBANK</td>
<td>12I5</td>
<td>Numbers of reaches subject to bank erosion, upstream to downstream</td>
</tr>
<tr>
<td>1210</td>
<td>PBANK(M,K) K = 1, N1 M = 1, NBANK</td>
<td>6F10.2</td>
<td>Cumulative distribution function for eroded bank material; K varies first, M varies upstream to downstream</td>
</tr>
<tr>
<td>139</td>
<td>BEROS(M) M=1,NBANK</td>
<td>6F10.2</td>
<td>Bank erosion rates ft³/day/mile in reaches LOCBER(M)</td>
</tr>
<tr>
<td>14</td>
<td>IDAYDT(M) NEWDT(M), M = 1,ICHGOT</td>
<td>16I5</td>
<td>Days at which time step changes, new time step in days</td>
</tr>
<tr>
<td>*15</td>
<td>DS(K),K=1,N1+1</td>
<td>6F10.4</td>
<td>Sediment sizes (mm) delimiting the N1 size intervals</td>
</tr>
</tbody>
</table>

Card types 16, 17, 18 are read for each of N computation points, I=1,N-1, downstream; types 20,21 are read for I=N

<p>| *1611 | RMILE(I) MA(I) | F10.2(1-10) I10(11-20) | River mile of computational point I Number of levels used to define cross section; MA(I) &lt; MAXMA |
| *1712 | STAGE(I,L) AREA(I,L) R1(I,L) B1(I,L) L=1,MA(Z) | F10.3(1-10) F10.3(11-20) F10.3(21-30) F10.3(31-40) | Reference elevation, ft Cross-sectional area, ft² Hydraulic radius, ft Surface width, ft (lowest to highest level, one level per card) |</p>
<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CDF(K), K=1,N1+1</td>
<td>Cumulative distribution function for bed sediment in the reach between points I and I+1; CDF(K) corresponds to DS(K)</td>
</tr>
<tr>
<td>NBEL(I), I=1,N-1</td>
<td>Number of elevations at which bed material changes in reach 1, I from downstream to upstream</td>
</tr>
<tr>
<td>THBED(I), I=1,N-1</td>
<td>Constant thickness (ft) of subsurface layers in reach I</td>
</tr>
<tr>
<td>PBED(I,K,J), K = 1,N1, J=1,NBEL(I), I=1,N-1</td>
<td>Probability distribution function for sediment in subsurface layer J of reach I</td>
</tr>
<tr>
<td>SSC(N,K), K=1,N1</td>
<td>Sediment concentration (ppm) by size fraction at the upstream boundary</td>
</tr>
<tr>
<td>IBLR</td>
<td>0 for no bed layer; 1 if bed layer is used instead of armoring procedure (cards 26 and 27 are skipped if IBLR=1)</td>
</tr>
<tr>
<td>IARMOR</td>
<td>0 for direct specification of armoring size; 1 otherwise</td>
</tr>
<tr>
<td>MIND</td>
<td>If IARMOR=0, number of smallest sediment size fraction in armor coat</td>
</tr>
<tr>
<td>QMAX</td>
<td>If IARMOR=1 and IQMAX=0, constant discharge used to determine armoring size, cfs</td>
</tr>
<tr>
<td>IQMAX</td>
<td>If IARMOR=1, 0 for determination of armoring size based on QMAX, 1 if based on local discharge</td>
</tr>
<tr>
<td>IACF</td>
<td>0 for no initial armored area; 1 if bed is armored initially</td>
</tr>
<tr>
<td>KARM</td>
<td>0 for specifying constant values of armoring coefficient (CARM); 1 for using Gessler relation; 2 for bed-load method; 3 for using both Geller's and bed-load methods; 4 for dune-height method; 5 for using both Gessler's and dune-height methods</td>
</tr>
<tr>
<td>C1</td>
<td>Coefficient for bed-load method</td>
</tr>
<tr>
<td>ARM(I,K)</td>
<td>Initial value of armor-covered bed area (fraction) for each sediment fraction K at reach I; and only if IACF=1</td>
</tr>
</tbody>
</table>

The input data structures for width changes, cutoffs and dredging are presently being revised. The user should therefore adopt IDREJ=ICHB=ICOFF=0 to suppress use of these features.

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITDAT</td>
<td>Date (day number) associated with the list of time dependent data to follow</td>
</tr>
</tbody>
</table>
TFREAD$^{16}$  F4.0 Water temperature (°F) on day ITDAT
QTRIB(L)  9F8.0 Tributary water discharges (cfs) on day ITDAT
L=1,NTRIB (recall QTRIB(1) = mainstem inflow)
YREAD  F8.0 Downstream water level on day ITDAT (ft)
QSUPS  F8.0 Upstream sedimentconc. inflow, ppm by weight
(used if INDSS=1, IUPS=0)

Notes

1. Used only if IBED=1.
2. The diagnostic messages are lengthy and of use only to the user who knows
   the detailed workings of the program.
3. NRES is the FORTRAN reference number of a sequential file onto which is
   written, without format, all results at each time step. This file can be
   used for off-line analysis of the computation. See Appendix E for the
   structure of this file.
4. If IDSWS=1, the program uses TLTM to compute the water surface elevation
   at the downstream boundary, for an imposed energy slope of STR (see Card
   5).
5. The Missouri River model uses $C_1 = 0.5$.
6. The Missouri River model uses STR=0.00189.
7. Card 6 is read only if IROCK=1.
8. AC has units of tons/day; BC is dimensionless. AD(1), BC(1) are read only
   if IUPS=1.
9. Card(s) read only if NBANK ≠ 0.
10. Cards 12 are read only if NBANK ≠ 0 and IBSED=1.
11. River miles increase from downstream to upstream, and must represent
    actual distances along the mainstem.
12. If AREA(I,MA(I)) is left blank, both the areas and hydraulic radii
    (average depths) will be calculated automatically by the trapezoidal
    rule. Otherwise the user must furnish consistent values.
13. Card(s) read only if IBED=1.
14. SSC(N,K) is not used, but must be read.
15. IALLUVIAL obtains time-dependent data by linear interpolation (in time)
    between successive data lists of type 27 cards. At least two type 27
    cards are required; ITDAT < 0 on the first, ITDAT > NT*360/NTY on the
    last.
16. TFREAD is used only if IDSWS=0.

64
APPENDIX B. ALLUVIAL MEMORY STRUCTURE

The general scheme used for memory allocation is described by Holly and Karim (1983), Appendix B. Changes to that structure are included in the updated table included herein.
<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Dimensions</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q</td>
<td>N</td>
<td>Water discharge @ each point</td>
</tr>
<tr>
<td>QTR</td>
<td>NTRIB</td>
<td>Tributary water discharge</td>
</tr>
<tr>
<td>LOCCTR</td>
<td>NTRIB</td>
<td>Location (node no.) of tributary</td>
</tr>
<tr>
<td>STAGE</td>
<td>N</td>
<td>Location (node no.) of tributary</td>
</tr>
<tr>
<td>VOLTN</td>
<td>N</td>
<td>Water surface elevation @ each point</td>
</tr>
<tr>
<td>REACH</td>
<td>NN</td>
<td>Depth of sediment entering mixing layer</td>
</tr>
<tr>
<td>DMZ</td>
<td>N</td>
<td>Reach length</td>
</tr>
<tr>
<td>MA</td>
<td>N</td>
<td>Depth of rock outcrop below initial bed</td>
</tr>
<tr>
<td>STAGE1</td>
<td>MAXNA</td>
<td>No. of definition levels for each section</td>
</tr>
<tr>
<td>BI</td>
<td>N MAXNA</td>
<td>Levels @ which section properties defined</td>
</tr>
<tr>
<td>SLI</td>
<td>NN</td>
<td>Width @ section definition levels</td>
</tr>
<tr>
<td>RI</td>
<td>N MAXNA</td>
<td>Initial bed slopes</td>
</tr>
<tr>
<td>AREA</td>
<td>N MAXNA</td>
<td>Hydraulic radius @ section definition levels</td>
</tr>
<tr>
<td>AREA1</td>
<td>N MAXNA</td>
<td>Cross-sectional area</td>
</tr>
<tr>
<td>RI</td>
<td>N MAXNA</td>
<td>Initial cross-sectional area, before updating</td>
</tr>
<tr>
<td>STAGE1</td>
<td>N MAXNA</td>
<td>Initial hydraulic radius, before updating</td>
</tr>
<tr>
<td>D50</td>
<td>NN</td>
<td>Initial section definition levels</td>
</tr>
<tr>
<td>D501</td>
<td>N</td>
<td>Particle size diameters defining size fractions</td>
</tr>
<tr>
<td>D50</td>
<td>NN</td>
<td>Median sediment size by reach</td>
</tr>
<tr>
<td>D501</td>
<td>N</td>
<td>CDF for initial bed sediment by reach</td>
</tr>
<tr>
<td>DMS</td>
<td>N</td>
<td>D50 by section, averaging of adjacent reaches</td>
</tr>
<tr>
<td>P</td>
<td>N</td>
<td>Particle size distribution by size fraction</td>
</tr>
<tr>
<td>D</td>
<td>N1</td>
<td>Geometric mean sediment size of each fraction</td>
</tr>
<tr>
<td>PT</td>
<td>NN N1</td>
<td>Sediment fraction in bed</td>
</tr>
<tr>
<td>PTT</td>
<td>NN N1</td>
<td>Initial sediment fraction in bed</td>
</tr>
<tr>
<td>TF</td>
<td>72</td>
<td>Temperatures @ which water viscosity defined</td>
</tr>
<tr>
<td>W</td>
<td>N1</td>
<td>Fall velocity for each size fraction</td>
</tr>
<tr>
<td>SSC</td>
<td>N N1</td>
<td>Susp. sed. conc by size interval</td>
</tr>
<tr>
<td>SSI</td>
<td>N1</td>
<td>Susp. sed. conc @ u/s section</td>
</tr>
<tr>
<td>PTTI</td>
<td>NN N1</td>
<td>Initial storage for PT</td>
</tr>
<tr>
<td>XAREA</td>
<td>N</td>
<td>Total cross-sectional area</td>
</tr>
<tr>
<td>R</td>
<td>N</td>
<td>Total cross-sectional area</td>
</tr>
<tr>
<td>B</td>
<td>N</td>
<td>Water at water surface</td>
</tr>
<tr>
<td>CTO</td>
<td>N</td>
<td>Mean sed. conc per volume</td>
</tr>
<tr>
<td>SF</td>
<td>N</td>
<td>Energy slope</td>
</tr>
<tr>
<td>ACF</td>
<td>N</td>
<td>Fraction of bed surface, which is armored</td>
</tr>
<tr>
<td>CIN</td>
<td>N</td>
<td>Initial values of mean sed. conc</td>
</tr>
<tr>
<td>FR</td>
<td>N</td>
<td>Friction factor</td>
</tr>
<tr>
<td>SG</td>
<td>N1</td>
<td>Specific wt by size fraction</td>
</tr>
<tr>
<td>VAV</td>
<td>N</td>
<td>Mean water velocity</td>
</tr>
<tr>
<td>SE</td>
<td>N</td>
<td>Energy slope</td>
</tr>
</tbody>
</table>

O indicates use of the array in the indicated subroutine.
<table>
<thead>
<tr>
<th>DW</th>
<th>DELDS</th>
<th>TOLED</th>
<th>CDEP</th>
<th>KA</th>
<th>BZ</th>
<th>TALWAP</th>
<th>VOLOUT</th>
<th>AKMILE</th>
<th>LLIIM</th>
<th>CLFT</th>
<th>DH</th>
<th>TI</th>
<th>TH</th>
<th>TM1</th>
<th>PTP</th>
<th>EL1</th>
<th>EL2</th>
<th>PTU</th>
<th>BM1</th>
<th>QST1</th>
<th>TDRLT</th>
<th>TRL1</th>
<th>AC</th>
<th>BC</th>
<th>LORR</th>
<th>BEAUS</th>
<th>NSEL</th>
<th>THERD</th>
<th>PBED</th>
<th>MAXBED</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
</tbody>
</table>

- Mean water depth
- Bed Elev^2 change due to suspended load
- Bed Elev^2 change due to total load
- Cumulative sum of TOLED
- Scour limit index
- Mixed layer thickness
- Bed Elev. in previous time step
- Depth of degr. in a time step
- River mile
- Index of exhaustion of sediment supply
- Computed Wt to satisfy stability criterion
- Computed cumulative water surface elevation change
- Mixed layer thickness, uncorrected
- Mixed layer thickness, corrected
- Temporary storage for TH
- Sediment size fractions, previous time period
- Intermediate working variable
- Intermediate working variable
- Updated sediment size distribution
- Mixed layer bottom elevation
- % of QS in each size fraction
- % of deposition in each size fraction
- Temporary storage for CDEP
- Mixed layer thickness, corrected
- Temporary storage for ACF
- Tributary sediment discharge
- Depth of sediment supplied by tributary
- Tributary sed. discharge size distribution
- Coefficient in sed. discharge rating curve
- Exponent in sed. discharge rating curve
- Location (reach no.) of bank erosion
- Bank erosion rate
- No. of sed. layers in a reach
- Thickness of sediment layers in a reach
- Fraction of sediment in each size interval, each layer
- No. of time intervals in which widths can change
- No. of time intervals in which cutoffs can occur
- Location of width changes (node number)
- Location of cutoffs (reach no.)
- No. of nodes undergoing width change in a time step
- No. of reaches undergoing cutoffs in a time step
- Size distribution of eroded bank material
- Updated size distribution of original material
<table>
<thead>
<tr>
<th>IDRT</th>
<th>NT</th>
<th>N</th>
<th>List of time steps in which dredging occurs, each time step</th>
</tr>
</thead>
<tbody>
<tr>
<td>IDRL</td>
<td>NT</td>
<td>N</td>
<td>List of locations (reach no.) of dredging, each time step</td>
</tr>
<tr>
<td>NDRL</td>
<td>NT</td>
<td>N</td>
<td>No. of dredging reaches in each time period</td>
</tr>
<tr>
<td>VDREJ</td>
<td>N</td>
<td></td>
<td>Volumetric rate of dredging &amp; each reach</td>
</tr>
<tr>
<td>DARM</td>
<td>N</td>
<td></td>
<td>Cumulative depth of degradation for armoring</td>
</tr>
<tr>
<td>RA</td>
<td>N</td>
<td></td>
<td>Pseudonym for R</td>
</tr>
<tr>
<td>BB</td>
<td>N</td>
<td></td>
<td>Total load deficit in previous time step</td>
</tr>
<tr>
<td>OSDP</td>
<td>N-1</td>
<td></td>
<td>Tributary water discharge</td>
</tr>
<tr>
<td>QTRIB</td>
<td>72</td>
<td></td>
<td>Bed elev. in current time step</td>
</tr>
<tr>
<td>TALGS</td>
<td>N</td>
<td></td>
<td>Previous value of armor-covered bed area for each sediment fraction k at reach l</td>
</tr>
<tr>
<td>ARMP</td>
<td>N</td>
<td></td>
<td>Initial value of armor-covered bed area for each sediment fraction k at reach l</td>
</tr>
<tr>
<td>ARM</td>
<td>NN</td>
<td></td>
<td>Sediment fraction in bed</td>
</tr>
<tr>
<td>PAC</td>
<td>N</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>T( )</th>
<th>MAIN</th>
<th>INFLOW</th>
<th>SEDBED</th>
<th>START</th>
<th>DACHAN</th>
<th>DADRED</th>
<th>EATPAD</th>
<th>RESIS1</th>
<th>TASEP</th>
<th>SECPAD</th>
<th>SLOPE</th>
<th>ARMOR</th>
<th>MECRID</th>
<th>SHIELD</th>
<th>TLAB</th>
<th>TERM</th>
<th>OUTPUT</th>
<th>CSSSEC</th>
<th>RESFUM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1106</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1107</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1108</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1109</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1110</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>133</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>134</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>127</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>141</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>142</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>18</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
APPENDIX C.

IALLUVIAL Source Listing
PROGRAM IALLUVIAL

DEFINITION OF VARIABLES

SECTION I, \( i = 1, n \)
SIZE FRACTION K, \( k = 1, n_1 \)
ELEVATION L, \( l = 1, n_2 \)
TIME INTERVAL IT, \( t_f = 1, n_T \)

IRES = INDEX VARIABLE TO INDICATE METHOD OF COMPUTING FLOW RESISTANCE
   IRES = 1 FOR IHR METHOD; IRES = 2 FOR A-K METHOD
ISED = INDEX VARIABLE TO INDICATE METHOD OF COMPUTING SEDIMENT DISCHARGE; ISED = 1 FOR IHR METHOD; ISED = 2 FOR EINSTEIN METHOD

INDEX = INDEX VARIABLE TO INDICATE NO. OF SUBSECTIONS USED IN BACKWATER CALCULATIONS (=1 FOR SINGLE CHANNEL; =2 FOR 2 SUBSECTIONS; =3 FOR 3 SUBSECTIONS)
INDEX = INDEX VARIABLE TO INDICATE WHETHER NO. OF SUBCHANNELS USED IN SEDIMENT CALCULATIONS IS THE SAME OR MORE THAN THAT USED IN BACKWATER CALCULATIONS (=0, SAME; =1, MORE)
INDSS = INDEX VARIABLE TO SPECIFY USTREAM SEDIMENT DISCHARGE; INDSS = 0 IF COMPUTED BY THE PROGRAM AS EQUAL TO THE TRANSPORT CAPACITY; =1 IF GIVEN AS INPUT
IDSW = INDEX VARIABLE TO SPECIFY DOWNSREAM WATER SURFACE ELEVATION; IDSW = 0 IF GIVEN AS INPUT; =1 IF COMPUTED INTERNALLY ASSUMING UNIFORM FLOW

N = NO. OF SECTIONS IN LONGITUDINAL DIRECTION
N1 = NO. OF SEDIMENT SIZE FRACTIONS
NIT = NO. OF TIME INTERVALS
NTRIB = NUMBER OF TRIBUTARIES
NBAK = NUMBER OF REACHES WITH BANK EROSION
IBED = INDEX VARIABLE TO INDICATE VERTICAL VARIATION OF BED-SAT. SIZE DIST. BELOW ORIGINAL CHANNEL BED; IBED = 0 FOR NO VARIATION; =1 FOR VARIATION.
NBED = NO. OF REACHES WHERE SEDIMENT-BED COMPOSITION VARIES IN VERTICAL DIRECTION
NTP = NO. OF TIME INTERVALS AT WHICH BED-ELEVATION CHANGES WILL BE PLOTTED AT A GIVEN SECTION
ICHB = INDEX VARIABLE FOR CHANGE IN CHANNEL WIDTH WITH TIME; ICHB = 0 FOR NO CHANGE; =1 FOR CHANGE.
ICOFF = INDEX VARIABLE FOR INCORPORATING CHANNEL CUTOFF AT SPECIFIED TIMES
IDREJ = INDEX VARIABLE FOR DREDGING; IDREJ = 0 FOR NO DREDGING; =1 FOR DREDGING.
IBUG = INDEX VARIABLE FOR PRINTING DETAILED OUTPUT FOR DEBUGGING; IBUG = 0 FOR NO PRINT; =1 FOR PRINTING
ILIMIT = LIMITING NUMBER OF VIOLATIONS OF SPECIFIED CRITERIA FOR RECOMPUTING BACKWATER PROFILE AND/OR SEDIMENT LOADS IN EACH TIME PERIOD
IEQ = INDEX VARIABLE TO INDICATE EQUILIBRIUM OR NON-EQUILIBRIUM SEDIMENT CALCULATIONS (IEQ = 1 FOR EQUIL., IEQ = 0 FOR NON-EQUIL.)
IDELT = TIME INTERVAL, DAYS
NTY=NO. OF TIME STEPS IN ONE YEAR (360 DAYS)
ITIME=TOTAL PERIOD, DAYS
IFLAG=INDEX VARIABLE TO INDICATE RECOMPUTATION OF BACKWARD
PROFILE (IFLAG=0: DO NOT RECOMPUTE; IFLAG=1: RECOMPUTE)
IFLAG1=INDEX VARIABLE TO INDICATE RECOMPUTATION OF SEDIMENT LOADS
IN EACH TIME PERIOD (IFLAG1=0, DO NOT RECOMPUTE;
IFLAG1=1, RECOMPUTE)

MA(I)=NO. OF INDEX ELEVATIONS FOR COMPUTING GEOMETRIC PROPERTIES
AT SECTION I
STAGE(I)=WATER SURFACE ELEVATION AT SECTION I, IN TIME INT. IT
REACH(I)=LENGTH OF REACH(FT.) BETWEEN SECTIONS I AND I+1
Q(I)=TOTAL DISCHARGE (CFS.) AT SECTION I, IN TIME INTERVAL IT
STAGE(1)(I,L), AREA(I,L), R1(I,L), B1(I,L)=STAGE, AREA, HYD. RADIUS, AND
W.S. WIDTH, RESPECTIVELY, OF THE WHOLE SECTION AT SECTION I,
ELEV. INDEX I
D50(I)=MEDIAN SEDIMENT SIZE (MM.) AT SECTION I
D65G(I)=D65 (MM.) AT SECTION I, SEGMENT J
DLIM(J)=MAX. SEDIMENT DEPTH (FT) AT START ON BED AT SECTION I, NEXT J
SEGMENT J
D(K), F(K), W(K)=SED. SIZE (MM.), POROSITY, AND FALL VELOCITY (FT/S),
RESPECTIVELY, FOR SIZE FRACTION K
PT(I,K)=SED. FRACTION FOR SECTION I, SEGMENT J, SIZE K, IN THE BED
PTT(I,K)=INITIAL VALUE OF PT(I,K)
PT1(I,K)=INITIAL STORAGE FOR PT(I,K)
TP(I), VIS(C)=ARRAY OF TEMPERATURE AND VISCOITY OF WATER,
RESPECTIVELY, AT INDEX I
TEMPF= TEMPERATURE (F) OF WATER IN TIME PERIOD IT
GS(K)=SPECIFIC WEIGHT (LBS/FT^3), FOR SIZE FRACTION K
SSC(I,K)=SED. CONCEN. (PPM) OF FRACTION K, AT SEGMENT J, SECTION I
SS1(K)=SSC(J,K)
B(I)=W.S. WIDTH
SF(I)=ENERGY SLOPE
CTO(I)=MEAN SED. CONCENTRATION PER VOLUME, AT SECTION I
PR(I)=FRICTION FACTOR
PSD(K)=FRACTION OF MATERIAL OF SIZE K IN THE DISCHARGE
RX(I)=HYD. RADIUS AT SECTION I, SEGMENT J
SRI(I)=CROSS SECTION AREA AT SECTION I, SEGMENT J
BB(I)=W.S. WIDTH
D50G(I)=SED. DIA. (MM.) AT SECTION I, SEGMENT J
QST(I)=TOTAL DISCHARGE OF SEGMENT J, AT SECTION I
VAV(I)=MEAN VELOCITY AT SECTION I, SEGMENT J
SE(I)=ENERGY SLOPE AT SECTION I, SEGMENT J
DW(I)=MEAN WATER DEPTH OF SEGMENT J, AT SECTION I
DELDB(I,K)=CHANGE IN BED ELEVATION AT SECTION I, SEGMENT J, FOR
SED. FRACTION K, DUE TO BED LOAD
DELD(I,K)=CHANGE IN BED ELEVATION AT SECTION I, SEGMENT J, FOR
SED. FRACTION K, DUE TO SUSPENDED LOAD
TDI(K)=CHANGE IN BED ELEVATION AT SECTION I, SEGMENT J, FOR
SED. FRACTION K, DUE TO SUSPENDED LOAD AND BED LOAD
CDEP(I)=SUM IN K AND TIME OF TDI(K)
CC(J)=SUM IN K OF TDI(I,K)
AREA1(I,K)=CROSS SECTIONAL AREA AT SECTION I, SEGMENT J, ELEV. IND. K
R11(I,K)=HYDRAULIC RADIUS AT SECTION I, SEGMENT J, ELEV. IND. K
BB11(I,K)=W.S. WIDTH AT SECTION I, SEGMENT J, ELEV. IND. K
XAREA(I), XAREAAL(I), XAAREA(I), XAAREA(I) =
DSOL (1), DSON (1), DSON (1) = SED.IIA. (MN.) FOR LEFT, RIGHT AND MAIN SUBSECTION, RESPECTIVELY

RKK1-RKK9 = COEFFICIENTS OF EQUATIONS (1) & (2) OF IIRK REPORT NO. 267

DYNAMIC ALLOCATION OF ARRAYS

ATTENTION: ARRAY T MUST ALWAYS BE DIMENSIONED T(MEMO)

DIMENSION T(20000), TITLE(15)
MEMO=20000

COMMON/DIMS/N,N1,NA,NA1,NAAMA,NN,NA1,IGN,NP1,STONA,ROBS,
NT3651, NT3652, NT413, NREK, NED, NAXED, NREX, NTP, NT1, IDREJ, NTY
COMMON/WATRES/PPC3, CPFU, PP, IRK, CPFREV, FF, CON3, CON4, CON5,
CALL STEPS (U)
TIME=TSTMS (INT1, INT2)
CPU=INT1
TIME2=0.

READ (5, 1000) TITLE
1000 FORMAT (15A4)
WRITE (6, 2000) TITLE
2000 FORMAT (11, '///', 10X, 70 ('!*'), '///', 10X, '!*', 60X, '!*', '///', 10X, '!*', 4X,
$ 15A4, 4X, '!*', '///', 10X, '!*', 60X, '!*', '///', 10X, 70 ('!*'), '///')

READ DIMENSIONING PARAMETERS

READ(5, 1001) N, N1, NT, MAAMA, NTREK, NEX, NED, NAXED, IDREJ, NTY
1, IDCUP
1001 FORMAT (13I5)
IF (IDCUP .LE. 0) IDCUP = 1
RKK1 = -2.2736
RKK2 = 2.9719
RKK3 = 1.0600
RKK4 = 0.2939
RKK5 = 0.9045
RKK6 = 0.1665
RKK7 = 0.0831
RKK8 = 0.2166
RKK9 = 0.0411
NN = N - 1
NMA = 4*NAXMA
NN1 = N*N1
NP1 = N1 + 1
NT3651 = 1
NT3652 = 1
IF (NTY .NE. 0) NT3651 = NT/NTY + 1
IF (NTY .NE. 0) NT3652 = NT/NTY + 2
NT1 = 1
IF (NTP .NE. 0) NT1 = NP1/NTP + 2

72
DYNAMIC ALLOCATION OF WORKING ARRAYS WITHIN ARRAY T

I1=1
I2=I1+N
I3=I2+N
I4=I3+N
I5=I4+N
I6=I5+N
I7=I6+N
I8=I7+N
I9=I8+N
I10=I9+N
I11=I10+N
I12=I11+N
I13=I12+N
I14=I13+N
I15=I14+N
I16=I15+N
I17=I16+N
I18=I17+N
I19=I18+N+1
I20=I19+N
I21=I20+(N+1)
I22=I21+N
I23=I22+N
I24=I23+N
I25=I24+N
I26=I25+N
I27=I26+N+1
I28=I27+1
I29=I28+N
I30=I29+N
I31=I30+N
I32=I31+N
I33=I32+N
I34=I33+N
I35=I34+N
I36=I35+N
I37=I36+N
I38=I37+N
I39=I38+N
I40=I39+N
I41=I40+N
I42=I41+N
I43=I42+N
I44=I43
I45=I44
I46=I45
I47=I46
I48=I47+N
I49=I48+N
I50=I49+N
I51=I50+N
I52=I51+N
I53=I52+N
I54=I53+N
I55=I54+N
156=155+N
157=156+N
158=157+N
159=158
160=159
161=160
162=161
163=162+N
164=163+N*N
165=164
166=165
167=166+N
168=167
169=168
170=169
171=170
172=171
173=172
174=173+N*N
175=174+N
176=175+N
177=176+N*N
178=177+N
179=178+N
180=179+N*N
181=180+N
182=181+N*N
183=182+N*N
184=183+N
185=184+N*N
186=185+N
187=186+N*NB
188=187+N*NB
189=188+N*NB*N
190=189+N*NB
191=190+N*NB*N
192=191+N*NB
193=192+N*NB
194=193+N*NB
195=194+N*NB
196=195+N*N
197=196+N*N
198=197+(N*N*N)*N*NB
199=198+NT3651
200=199+NT3651
201=200+NT3651*N
202=201+NT3651*N*N
203=202+NT3651
204=203+NT3651
205=204+N*NB
206=205+N*N
207=206+N*NB
208=207+N*NB
209=208+N*NB
210=209+N*NB
211=210+N*N
212=211+N*N
VERIFICATION OF SUFFICIENT MEMORY

IF (IEND.LT.MEMO) GO TO 10
CALL ERROR (IEND, MEMO, KA, NL, NT, MAXMA, NOBS, NX, IGR)

MEMORY O.K. TRANSFER CONTROL AND ARRAY ADDRESSES TO SMAIN

WRITE (6, 2002) IEND, MEMO
2002 FORMAT (120, 'MEMORY USED =' // 18, ' WORDS', 3X, 'MEMORY AVAILABLE =' // 18)
CALL SMAIN (TITLE, I (1), T (112), T (13), T (15), T (17),
1 T (19), T (110), T (114), T (113), T (114), T (115), T (16),
2 T (117), T (119), T (120), T (121), T (122), T (123), T (124), T (125),
3 T (126), T (129), T (130), T (131), T (132), T (133), T (134),
4 T (135), T (136), T (137), T (138), T (139), T (140), T (14), T (127),
5 T (141), T (142)
CALL SMAIN1 (T (147), T (148), T (149), T (150), T (151), T (152),
6 T (153), T (154), T (155), T (156), T (157),
7 T (162), T (163), T (164),
8 T (173), T (174), T (175), T (176), T (177), T (178), T (179),
9 T (180), T (181), T (182), T (183), T (184), T (185), T (186), T (187), T (188),
10 T (189), T (190), T (191), T (192), T (193), T (194), T (195), T (196), T (197),
11 T (198), T (199), T (200), T (201), T (202), T (203), T (204), T (205),
12 T (206), T (207), T (208), T (209), T (210), T (15), T (16))
TIME=STEPS (IN1, IN2)
TIME2=(IN1-CPU)/38400.*TIME2
WRITE (6, 500) TIME2
500 FORMAT ('0', 6X, 'CPU FOR MAIN = ' // 5X, F10.6)
STOP
END

FIRST CARD OF SUBROUTINE SMAIN ###############

--------------
SUBROUTINE SMAIN
--------------

SUBROUTINE SMAIN (TITLE, VOLIN, REACH, DMZ, STAGE, Q,
1 MA, TALWG, BT, SL1, R1, AREA, AREA1, RI, STAGE1, DS, D50, CDF, DNS, P, D,
2 PT, PTT, TF, W, SSC, SST, PT1, AREA, R, D, CTO, SF, ACP, CIR, PR, GS, QSDP,
3 TTRIB, TALWG, ARM)

TIME=STEPS (IN1, IN2)
CPU=IN1
INTEGER ICODE, 'IAL1'/
LOGICAL SECCAL

DIMENSION VOLIN (N, N1), REACH (NN), TITLE (15), DMZ (N),
1 STAGE (N), QA (N), MA (N),
2 BT (N, MAXMA), SL1 (NN), R1 (N, MAXMA), AREA (N, MAXMA), AREA1 (N, MAXMA),
3 MAXMA, RI (N, MAXMA), STAGE1 (N, MAXMA), DS (N, TF1), D50 (NN), CDF (N, TF1),
4 DNS (N), P (N1), D (N1), PT (N1, N1), PTT (NN, N1), TF (72),
5 VISC (72), W (N1), SSC (N, N1), SST (N1, N1), PT1 (NN, N1), XAREA (N),
6 R (N), D (N), CTO (N), SF (N), ACP (N), CIR (N), PR (N), TALWG (N),
7 YAV (N), QSDP (N), TALWG (N),
8 SE (N), DW (N), DELDS (N, N1)
DIMENSION IDD1D (N, N1), CDF (N, KA (N, N1), TALWG (N),
1 VOLOUT (N), RAILLE (N), LLIM (N), DELT (N),
2 N1, DPH (N), IDAYDT (100), HEADT (100),
3 NI (N, N1), TH (N), TL (N), TP1 (R, N),
4 SL1 (N), ELZ (N), PT1 (N, N1), DL (N), PQS (N, N1), DS (N, N1),
DATA VISC / 1.92, 1.89, 1.85, 1.82, 1.79, 1.76, 1.72, 1.69, 1.66, 1.64, 
 1.61, 1.58, 1.55, 1.53, 1.50, 1.48, 1.45, 1.43, 1.41, 1.39, 1.37, 1.34, 
 1.32, 1.30, 1.29, 1.25, 1.23, 1.21, 1.19, 1.17, 1.16, 1.14, 1.13, 
 1.11, 1.10, 1.10, 1.09, 1.08, 1.07, 1.05, 1.04, 1.03, 1.02, 1.01, .999, .986, .974, .961, 
 .949, .938, .926, .915, .904, .893, .883, .873, .862, .852, .843, .833, 
 .824, .814, .805, .796, .788, .780, .771, .762, .754, .746, .735, .731, 
 .723, .716 /
COMMON/DIMS/N,N,1,NT,SI,MAXIP,NN,MAXD,NA,IGF,NT1,NTONX,NOBS,
 1 NT3651,NT3652,NT3653,NT3654,NT3655,NT3656,NT3657,NT3658,NT3659,
 1 IDELT,VISLT,TIME,DIAG,IFLAG,IT,IDR,INDS,IPG,1,
 1 IMIT,AM,LIK,THSL,ISBD,ALP,DATA,C2I,C22,PA1X,IPR,IPRINT,
 2UFP,STAM,CPDUG,ELUG,ICH,COFF,VISLOG,THETA
COMMON/DRFJK/DRFJK
COMMON/BANK,ISBD,QSBD,UPS
COMMON/AR1,ILRA
COMMON/HD/1,SH,BSL
COMMON/CPUTIME,TIME(1:10),CULD(1:10)
COMMON/DSB/D,DSH,DSF,DSH,DSN,DSN,DSN,DSN,DSN,DSN
COMMON/DSB/1,DSB,DSB,DSB,DSB,DSB,DSB
COMMON/NAMLIST/DSB,ISBD,ISBD,ISBD,ISBD,ISBD,ISBD,ISBD
COMMON/NAMLIST/ISBD,ISBD,ISBD,ISBD,ISBD,ISBD,ISBD

DO 1999 I=1,18
  TIME1(I)=0.
  CALDF(I)=0.
1999 CONTINUE

FUNITS=62.4*2.65/2000.
CONVOL=0.0
  TIME=TSTMPS(INT1,INT2)
  TIME1(I)=(TIME-CPU)/36400.+TIME1(I)
  CALDF(I)=CALDF(I)+1.
RETURN

***********************************************************
ENTRY SMAIN(VAV,SE,DE,DELUS,DELUD,DEEP,KA,BZ,TAL*GP,
 4 VOLUT,AMILE,ILLI,DELT,DE,
 5 TL,TI,TI2,TPF,ELI,EL2,TL,BRP,POS,PDN,DEEP,TEB,ACF,
 6 QTR,STR,TETR,LOC,TAB,AC,BC,LOCBER,BEROS,NELL,THELD,BED,
 7 ICH,LNCOFF,LNCGL,LNCGL,LNCFF,PEAK,PTA,IDR,TIDR,
 8 NDRL,VRDJ,DAM,AIM,PAJ)
***********************************************************

DATA INPUT

IPR=INDEX VARIABLE TO INDICATE NODE OF CALCULATING PR. FACTOR
IPR=0 FOR CONSIDERING PR.FR. AS FUM. OF QS IN CURRENT PERIOD
IPR=1 FOR CONSIDERING PR.FR. AS FUM. OF QS IN PREVIOUS PERIOD
IDIA=INDEX VARIABLE TO INDICATE PROCEDURE OF FINDING D50 AT
A SECTION FROM ADJACENT MACX VALUES
IDIA=0 FOR CALCULATING D50 BY AVERAGING ADJ. MACX VALUES
IDIA=1 FOR CALCULATING D50 AS THE U/S REACH VALUE
IUF=INDEX VARIABLE TO INDICATE PROC. TO SPECIFY W/S ELEVATION
AT THE MOST D/S SECTION
IUF=0 FOR SPECIFIED W/S ELEV. AT THE MOST D/S SECTION
IUF=1 FOR CALCULATING W/S ELEV. FOR UNIFORM FLOW
IMPR=INDEX VARIABLE TO INDICATE FREQUENCY (NO. OF TIME INTERVALS)
OF PRINTING RESULTS
KDIA=INDEX VARIABLE FOR CONSIDERING ARMORING EFFECT IN CALC. D50
KDIA=0 FOR NOT CONSIDERING ARMORING EFFECT
KDIA=1 FOR CONSIDERING ARMORING EFFECT
IGR=INDEX VARIABLE TO INDICATE PLOTTING OPTION
IGR=0 FOR NO PLOT; IGR=1 FOR PLOT
ABSOLUTE VALUES OF BED W/S ELEVATIONS; =1 FOR PLOTTING
INCREMENTAL CHANGES IN BED AND W/S ELEVATIONS
IROCK=INDEX VARIABLE FOR LIMITING DEGRADATION DUE TO ROCK OUTCROP
IROCK=0 FOR NO ADJUSTMENT DUE TO ROCK OUTCROP
IROCK=1 FOR ADJUSTMENT DUE TO ROCK OUTCROP

INPUT=INDEX VARIABLE TO DESCRIBE DATA INPUT
INPUT=0 FOR SIMPLIFIED DATA INPUT FOR THE NO. RIVER
INPUT=1 FOR GENERAL DATA INPUT

READ CONTROL VARIABLES

TIME=TSTMPS(1INT1,INTZ)
CP0=INT1
KDIA=0
IDIA=0
IUF=0
READ(5,2) ITRK, IROCK, ICSGLT, IUPS, IBUG, ICHE, ICOFF, ITMX, NRES,
@LASTk, IDSWS, NRESTA, NEWSA
IF(LAST1.LE.0) LAST1=99999
IF(NRES.GT.0. AND. NRESTA.EQ.NRES) WRITE(6,2004)
2004 FORMAT(' ERROR: NRES MUST BE DIFFERENT FROM NRESTA!')

OPEN RESULTS FILE WRITE GENERAL DATA ON IT

IF (NRES.GT.0) WRITE(NRES) ICODE
IF (NRES.GT.0) WRITE(NRES) TITLE
IF (NRES.GT.0) WRITE(NRES)
IN, NT1, NT2, NAKRA, NN, BCT, NA, IGR, ITMX, NTMX, NRES,
2 NT3651, NT3652, NT1B, NEARK, ISED, NAED, ND, NBED, NT2, N1, NDK, NTY
IRES=1
ISED=1
INDEX=1
INDEX=0
INDSS=1
ILIMIT=5
IEQ=1

READ CALIBRATION PARAMETERS AND DIVERSE PHYSICAL PARAMETERS

READ(5,5) ALFA, BETA, C21, C22, PM1X, CAM1
READ(5,3) STR, CRITK, THETA, ADJRE
IF(THETA.LE.0.) THETA=0.5
SF(1)=STR
IF(IROCK.EQ.1) READ(5,5) (DLZ(I), I=1, N)
IDELT=NTY
IFR=0
WRITE(6,54) N,NI1,ST,MAANA,NTRIB,NEANK,IBED,HABED,IBRSJ,NTY
1,ICUP
WRITE(6,55) IPR,INROCK,ICGDT,IPWS,IBUG,ICHG,COFF,ITMAX,AMES,
*LASTAM,IDSWS,ARESTA,NEWSTA
WRITE(6,31) ALFA,BETA,CZ1,CZ2,PMIA,CARR
WRITE(6,29) STR,CNTER,THETA,ADEUE

CALL TRIB TO READ TRIBUTARY DATA

CALL TRIB(QTK,QSTF,IPBLTR,LOCTR,PTRIB,AC,SC,Q,LOCBE,SROS,
&PBANK,CT0,INPUT)

NN1=N1+1

READ TIME STEP CHANGES

IF (ICGDT.GT.0) READ (5,1000) (IDAYDT(I),NEWDT(I),I=1,ICGDT)
1 IF (ICGDT.GT.0) WRITE (6,2000) ICHGDT, (IDAYDT(I),NEWDT(I),
1 I=1,ICGDT)

1000 FORMAT (1615)
2000 FORMAT (/T10,15,'TIME STEP CHANGES:','T41,'TIME DELT ETC',' ,
1 /,T10,24(1E-), (T40,1615))

READ STANDARD SEDIMENT SIZES

READ (5,5) (DS(N),N=1,NN1)
WRITE (6,20)

READ SECTION DATA, COMPUTE RIVER MILES, AREAS, HYDRAULIC
RADII (LAST AREA=0.0 INTERPRETED AS REQUEST TO CALCULATE
ALL AREAS)

DO 100 I=1,N
4110 FORMAT (F10.2,11)
READ (5,4110) RMILE(I),MA(I)
1 IF (I.GT.1) REACH(I-1)=RMILE(I)-RMILE(I-1))*5280.0
MA=MA(I)
1 IF (MA.GT.MAXMA) CALL ERROR (1,MM ,MAXMA)
WRITE (6,8) I,RMILE(I),MA(I)
READ (5,4) (STAGE(I,L),AREA(I,L),R1(I,L),
1 B1(I,L),L=1,MM)
3(I)=B1(I,MM)
SECCAL=.TRUE.
IF (AREA(I,MM).NE.0.) SECCAL=.FALSE.
DO 312 L=1,MM
1 IF (L.EQ.1 OR .NOT.SECCAL) GO TO 750
AREA(I,L)=AREA(I,L-1)*0.5*(B1(I,L-1)+B1(I,L))
1 *(STAGE(I,L)-STAGE(I,L-1))
R1(I,L)=AREA(I,L)/B1(I,L)
750 AREA(I,L)=AREA(I,L)
R1(I,L)=R1(I,L)
312 CONTINUE
TALWP(I)=STAGE(I,L)
TALWG(I)=TALWP(I)
SP(I)=STR
WRITE (6,16) (STAGE(I,L),AREA(I,L),R1(I,L),B1(I,L),L=1,MM)
IF (I.EQ.N) GO TO 100
READ SEDIMENT CHARACTERISTICS FOR REACH

READ (5,5) (CDF(K),K=1,NN1)

COMPUTE D50 FROM GIVEN CDF

DO 25 K=1,NN1
   IF(CDF(K) .GT. 0.5) GO TO 25
25 CONTINUE
K=NN1
D50(I) = DS(K-1) + (DS(K) - DS(K-1)) * (0.5 - CDF(K-1)) / 
        (CDF(K) - CDF(K-1))

TRANSLATE 'D50' FROM REACH TO SECTION

IF (I .NE. 1) DMS(I) = (D50(I) + D50(I-1)) / 2.0
IF (IDIA .EQ. 1) DMS(I) = D50(I)
DO 108 K=1,N1
   F(K) = 0.40
   D(K) = SQRT(DS(K) * DS(K+1))
   PT(I,K) = CDF(K+1) - CDF(K)
   EFF(I,K) = PT(I,K)
108 CONTINUE

WRITE (6, 1490) I, D50(I)
     (CDF(K), K=1,NN1)
WRITE (6, 61) (PT(I,K), K=1,N1)

WRITE RIVER MILES ON RESULTS FILE

IF(NRES .GT. 0) WRITE(NRES) (MILE(I), I=1,N)
DMS(1) = D50(1)
DMS(N) = D50(N-1)

GENERATE STANDARD TEMPERATURES FOR VISCOSITY TABLE

N3=72
TF(I) = 32.0
DO 203 I=2,N3
   TF(I) = TF(I-1) + 1.0
203 CONTINUE

CALL SEDBED TO READ DATA ON VERTICAL VARIATION OF BED
SEDIMENT COMPOSITION

IF (IBED .EQ. 1) CALL SEDBED (REAM,IBED,PBRED)

PRINT DIVERSE GENERAL DATA

WRITE (6, 44) (REACH(I), I=1,N1)
IF (IHOOK .EQ. 1) WRITE (6, 30)
IF (IHOOK .EQ. 1) WRITE (6, 33) (DS(I3), I3=1,N)
WRITE (6, 48) (F(K), K=1,N1)
WRITE (6, 79) (DS(K), K=1,NN1)
WRITE (6, 62)
WRITE (6, 37) (D(K), K=1,N1)
WRITE (6, 121)
WRITE (6, 33) (TF(I), I=1,N3)

79
CONVERSION OF VISCI TO (SC. FT./S)

DO 935 I=1,N3
VISCI(I)=VISCI(I)*1.E-5

CONVERSION OF SED. SIZE FROM NA TO PT.

DO 111 K=1,NN1
IF (K.NE.NN1) D(K)=D(K)/304.8
DS(K)=DS(K)/304.8
DO 814 I=1,N
D50(I)=D50(I)/304.8
IF (I.EQ.N) GO TO 814
D50(I)=D50(I)/304.8
CONTINUE

INITIAL SUBROUTINE CALLS TO TRANSFER ARRAY ADDRESSES

CALL DAWATP (REACH, STAGE, 2, TALNG, J, XAREA, R, B, CTO, SF, PR, ACF, 1 LOCTR, TR, LNS)
CALL DARESL (Q, DMS, D, XAREA, R, CTO, SF, ACF, CIN, H)
CALL DATRAS (D50, D, PT, SSC, R, CTO, SF, P, PNS, PDN, FAC)
CALL DASECP (STAGE, NA, STAGEBL, B1, R1, AREA, XAREA, R, B, TALNG)
CALL DASLOR (REACH, P, D, PT, T1, SSC, CTO, SF, ACF, R, B, D)
1 DELDS, TDEL, CDEP, NA, VOLOUT, DELT, EQS, PDN, D50, CDEP1, TB, ACF1,
2 LOCTR, STR, PTR1, TDEL1, LOCDEP, DEP, PDNS, PDNS1, Q, PTA, D50,
3 TBED, NEEL, T5, D50, TAV, T5
CALL DARAUS (VOLIN, TALNG, D50, D, PT, T1, SSC, CTO, SF, ACF, R, B,
1 TDEL, VOLOUT, LLIN, T1, T1, DELT, T1, T1, TPT, EL, TEL, T5, PTU, B1, TB, T5,
2 TBED, NEEL, CDEP, 0)
CALL DAARMO (ACF, ACF1, PT, D, E, CDEP, TPT, VOLOUT, R, SF, D50, T5,
1 T5, NEEL, T1, T5, P5, Q, R, D50, T5, A5, FAC, T1, 2
3 ARM)
CALL DASHL
IF (ICH8.EQ.1.0) CALL DACHAN (ICH8, ICH8, ICH8, ICH8, 6 ICH8, REACH, B, AREA, SCHL, ICH8, SCI, MA, R1)
IF (IDREJ.EQ.1) CALL LAARBL (LDRK, LDRK, NEEL, VDREJ, REACH, B, TALNG, 6 CDEP, AREA, R1, NEEL, T5, T5, PT, T5, P, D50, D50, EL, T5, MA)
ITIME=0
KYR=0
IT=-1

INITIALIZES ARRAYS FROM RESTART FILE

IF (NRESTART.LT.0) GO TO 355
READ (NRESTART, END=840) NCODE
IF (ICODEE.EQ.NCODE) GO TO 356
READ (NRESTART, END=840) TITLE
READ (NRESTART, END=840)
READ (NRESTART, END=840)
816 READ (NRESTART, END=840) IT, ITIME, IDAY, KYR, VOLUME, CUMVOL
IF (NRES.GT.0) WRITE (NRES) IT, ITIME, IDAY, KYR, VOLUME, CUMVOL
DO 820 I=1,N
II=I-1+N

80
READ (NRESTART, FRM=40) B (I1), STAG (I1), Q (I1), X (I1), VAV (I1),
                  CTO (I1), QSTR (I1)
IF (NRES.GT.0) WRITE (NRES) B (I1), STAG (I1), Q (I1), X (I1), VAV (I1),
                  CTO (I1), QSTR (I1)
IF (I1.LE.N) QSTR (I1) = CTO (I1) + CTO (I1)
IF (I1.EQ.N-1) QSTR (I1) = 0.0 - CTO (I1)
CSTR (I1) = STAG (I1) - TAIWOH (I1)
READ (NRESTART, END=840) (PT (I1, K), K=1,N1)
DO 818 K=1,N1
   PTH (I1, K) = PT (I1, K)
   ARM (I1, K) = ACF (II)
   ARMP (I1, K) = ARM (II, K)
818 CONTINUE
IF (NRES.GT.0) WRITE (NRES) (PT (II, K), K=1,N1)
820 CONTINUE
READ (NRESTART, END=840) (QTR (L), L=1,NTRIB), (QSTR (L), L=1,NTRIB)
IF (NRES.GT.0) WRITE (NRES) (QTR (L), L=1,NTRIB), (QSTR (L), L=1,NTRIB)
IF (ITIME.EQ.NREWST) GO TO 376
WRITE (6, 2002) ITIME, NRESTART, TITLE
2002 FORMAT ('///', I1, 'INITIAL CONDITION LOADED AT ITIME=', ITIME, ' FROM FILE',
               'I3,3H,8,'15A14,2H.,//',)
GO TO 850
840 WRITE (6, 2003) NRESTART, NEWSTA, ITIME
2003 FORMAT ('///', I1, 'ERROR READING RESTART FILE', ITIME, ' NEWSTA= ', NEWSTA, ' //',)
850 IPRINT=2
IT=0
CALL OUTPUT (K, IIL, VOLUME, CURVOL, DMS, VAV, B, SE, CTO,
                      STAG, TAIWOH, QSTR, QSUP, QLIM, VOLOUT, VNLIME, V, STAG,
                      DH, ACF, NRES, PI, 2TR, QSTR)
C
BEGIN LOOP ON TIME STEPS
C
855 NEXTP=INPR
IPLAG=0
MDT=1
IFLRN=0
TIMEI=0.
TIME=TSNPS (INT1, INT2)
CPUT=INT1
5000 IT=IT+1
IF (IT.GT.M1) GO TO 748
C
CHANGE TIME STEP IF APPROPRIATE
C
IF (ITIME.LT.IDAYDT (MDT)) GO TO 360
IDELT=NEWDT (MDT)
MDT=MDT+1
WRITE (6, 2001) IDELT, ITIME
2001 FORMAT ('///', I20, 'TIME STEP CHANGED TO', I5, ' DAYS AT ITIME=', ITIME)
860 IF (IT.GT.0) ITIME=ITIME+IDELT
IF (ITIME.GT.LASTIT) GO TO 748
C
LOAD NEW DISCHARGES, TEMPERATURE, AND D/S STAGE (IF IBSSW.NE.1)
CALL INFLOQ (Q, TR, LOCIR, TEMP, STAGE, QSUPS, QTRIM)

COMPUTE NEW TRIBUTARY SEDIMENT LOADS
CALL TRIBQS (QSUPS)

LOAD 0/S BOUNDARY CONDITION
IF (IDSNW.EQ.1) CALL START (STAGE, Q, STAGE1, DSS, B, XAREA, MA, B1, 
& K1, REACH, SF, AREA, TALWG)  
KDREJ=0

CALL DREDGE TO CALCULATE DREDGING
IF (IDREJ.EQ.1 .AND. IT.GT.0) CALL DREDGE

CALCULATE VISCOSITY FROM GIVEN TEMPERATURE IN EACH TIME PERIOD
DO 900 I=1,N3
IF (TEMP.LE.TF(1)) GO TO 910
900 CONTINUE
CALL ERRORS (IT, ITIME, TEMP, TF(N3))
I=N3
910 VIS=VISC(I)
VISLOG=A LOG10 (VIS)

ESTIMATING FALL VELOCITY BY BUEY EQN.
DO 214 K=1,N1
F11=.36*VIS**2/(32.2*2*(K)**3*1.65)
F1=SQRT(2.0/3.0+F11)-SQRT(F11)
W(K)=F1*SQRT(1.65*32.2*D0(K))
IF (IBUG.EQ.1) WRITE (6, 34) K, W(K)
34 FORMAT(10X,'W(',I2,')=',F10.4,*(FT/S)*)
214 CONTINUE

NOTE OLD DEPTHS FOR DOWNSTREAM BOUNDARY CONDITION
COMPUTATION BY CHARACTERISTICS
IPRINT=0
IF (IT.LE.2) IPRINT=2
IF (ITIME.EQ.360) IPRINT=2
IF (ITIME+IDELT.GT.LASTIM.OG.LT.EQ.NT) IPRINT=2
IF (ITIME.LT.NEXTP) GO TO 162
NEXTP=ITIME+INPK
IPRINT=2

BEGINNING OF ITERATION LOOP WITHIN TIME STEP

102 ITERDT=0
TIME=TIMPS (INT1, INT2)
CPU1=INT1
690 ITERDT=ITERDT+1
DELDZM=0.0

CALL TO WATPRO FOR BACKWATER COMPUTATION IN EACH TIME STEP
CALL WATPRO
CALCULATION OF MEAN FLOW PROPERTIES AT EACH SECTION

DO 700 I=1,N
VAV(I)=Q(I)/XAREA(I)
SE(I)=FR(I)*((VAV(I)**2)/(3.0*32.2*R(I)))
DW(I)=R(I)**2
TALWG(I)=TALWG(I)
CONTINUE

CALL TO LOAD FOR SEDIMENT CONTINUITY COMPUTATION

IF (IT.EQ.0) GO TO 713
CALL LOAD

MODIFICATION OF SECTION PROPERTIES AFTER SEDIMENTATION IN EACH TIME PERIOD

CALL QUSUSBC(DMS,XAREA,VAV,FR,STAGE,DMZ,QST,ACF,1)
DO 726 I=2,N
TALWG(I)=TALWGP(I)-(REACH(I-1)*VOLOUT(I-1)+REACH(I)*VOLOUT(I))/
(REACH(I)+REACH(I-1))
CONTINUE

TALWG(I)=TALWGP(I)-1.5*VOLOUT(I)+0.5*VOLOUT(2)
TALWG(I)=TALWGP(I)-VOLOUT(I)

MODIFICATION BY CHANGING INDEX ELEVATIONS OF X-SECTIONS
TAKING RACK OUTCROP INTO ACCOUNT.

DO 1110 I=1,N
IF (ILOCK.NE.0) TALWG(I)=MAX(TALWG(I),STAGE(I,1)-DMZ(I))
DELDZ=ABS(TALWG(I)-TALWGP(I))
IF (DELDZ.LE.DELDZ) GO TO 4560
IREACH=I
DELDZ=DELDZ
CONTINUE

RECALCULATION OF SEDIMENT DIA. AFTER SEDIMENTATION IN EACH ITERATION

IF (I.EQ.N) GO TO 1110
CDF(I)=0.0
DO 1112 K=2,N+1
IF (IBLR.EQ.0) CDF(K)=CDF(K-1)+PT(I,K-1)
IF (IBLR.EQ.0) CDF(K)=CDF(K-1)+PAC(I,K-1)
IF (CDF(K).GE.0.50) GO TO 1113
CONTINUE

DSO(I)=DS(K-1)+(0.50-CDF(K-1))/(CDF(K)-CDF(K-1))
*DSO(I)=DS(K)
DSOAM=DSO(I)*304.8
IF (IBUG.EQ.1) WRITE(6,1111) I,DSOAM,(PT(I,K),K=1,N)
1111 FORMAT(5X,'I4,1X,13,2X,D9.5,2X,PT: ',9F9.5)
1110 CONTINUE

TRANSLATE DSO(I) AND ACF(I) FROM REACHES TO SECTIONS

IF (IBUG.EQ.1) WRITE(6,1111) (ACF(I),I=1,N)
FORMAT (GX,'ACF : ',E12.3)
ACF(N)=ACF(N-1)
C
DWP=DW(N)
C
STAGE=NSTAGE(N)
DMS(1) =D50(1)
DMS(N) =D50(N-1)
ACFI=ACF(1)
DO 731 I=2,N
IF (I.LE.N) GO TO 715
DMS(I) =(D50(I)+D50(I-1)) /2.0
IF (IDIA.EQ.1) DMS(I) =D50(I)
ACPI=ACF(I)
ACPI=0.5*(ACF(I-1)+ACF(I))
ACF(I-1)=ACPI
731 CONTINUE

C
C ITERATION MANAGEMENT
C
IF (ITERDT.LT.ITERA.AND.DELEN.GT.CRITER) GO TO 690
TIME=TIME+(INT1+INT2)/100.
ITERN=ITERN+ITERN
C
C LAST ITERATION'S VALUES ADOPTED AS FINAL FOR TIME STEP
C
IF (ITERDT.GT.1) WRITE(0,3000) ITERDT,IT,ITIME,REACH,DELEN,CRITER
3000 FORMAT(T5,I5,11HGLOBAL ITERATIONS, IT=',I5,11HTIME=',F10.3,
1 1HCRITICAL POINT NO. ',I5,11HLAST CHANGE=',F10.3,
2 1HCRTIERION=',F10.3)
VOLUME=0.0
DO 725 I=1,N
IF (I.LT.N) QSDP(I)=CIO(I+1)-CIO(I)
IF (I.EQ.N) QSDP(I)=0.0-CIO(I)
CDP(I)=CDEP(I)
IF (I.LT.N) VOLUME=VOLUME+REACH(I)*0.25*(B(I)+B(I+1))
1 *(TALWG(I)-TALWG(I)+TALWG(I+1)-TALWG(I+1))
TALWGP(I)=TALWG(I)
IF (I.EQ.N) GO TO 725
DO 725 K=1,N
FTP(I,K)=FT(I,K)
ARM(I,K)=ARM(I,K)
725 CONTINUE
VOLUME=CURVOL+VOLUME*(1.-F(I))
VOLUME=VOLUME*PUNITS*(1.-F(I))
CALL OUTPUT(KIN,DAY,VOLUME,CURVOL,DMS,VAV,SE,CIO,
1 STAGE,TALWG,QSUPS,LLIM,VOLOUT,RAIL,STAGE,
2 DH,ACF,ATLS,FT,QR,TSTR)
IF (IT.EQ.0) GO TO 4999
C
C CALCULATING MAX. VALUE OF TIME INTERVAL WHICH WILL NOT VIOLATE
C SEDIMENT CONTINUITY THROUGH REACHES
C
4999 IF (IT.EQ.0) GO TO 5000
DMN=DELT(I,1)
IF (IT.EQ.1) NVI=0
IF (IT.EQ.1) VDEGR=0
DO 753 I=1,N
CONN=REACH(I)*((B(I)+B(I+1))/2.0* (1.0-F(I)))
VDEGR = VDEGR + VOUT(11) * COR
DO 753 K1 = 1, N1
DINT = FLOAT(1DELT)
IF (DINT.GT.DELT(11,K1)) NV1 = NV1 + 1
IF (DELT(11,K1).GE.DIN) GO TO 753
DIN = DELT(11,K1)
753 CONTINUE

VT1 = VDEGR *_UNITS_/ (1TIME/365.0)
VT2 = CUMVOL *_UNITS_/ (1TIME/365.0)
VT3 = VOLUME *_UNITS_/ (1DELT * 365.0)
IF (IPRINT.LT.2) WRITE (6, 43) DMIN, NV1, VDEGR, VT1, CUMVOL, VT2, VT3
FORMAT (/ 15X, '(*')/, 15X, '**', 66X, '*', '*\', 15X, '*', '*', 63X, '*\',
* 'VIOLATE', '8X, '*\', 15X, '*', '*', 3A, 'SEDIMENT CONTINUITY IS',
* '23X, '*\', 15X, '*', '*', 25X, 'DAYS', '28X, '*\', 15X, '*', '*',
* '28X, 15X(*-1), 75X, '*\', 15X, '*', '*', 20X, 'TOTAL NO. OF VIOLATIONS =',
* 'I4, 19X, '*\', 15X, '*', '*', 66X, '*\', 15X, '*\',
* '3X, 'DEGR.VOL. (SED.CONT.) =', 12X, 'TCS', 'CFT', '(', 31.2X,
* 'TONS/YR') 'T', 35X, '*\', 15X, '*\',
* '3X, 'DEGR.VOL. (PROFILE) =', 12X, 'TCS', 'CFT', '(', 31.2X,
* 'TONS/YR') 'T', 35X, '*\', 15X, '*\',
3 'CURRENT SCOUR RATE =', 12X, 'TCS', 'TONS/YR', 'T55', '*\', 15X
* '15X, '70 (1HR) ', //
4998 GO TO 5000
748 IT = IT - 1
TIME = TIMES (INT1, INT2)
TIMET = (INT1 - CPU TIME) / (35400. * IT)
TIMEI = TIME / (ITER * 364.)
TIMES = TIMEI / (ITER * 11)
WRITE (6, 7)
749 FORMAT (11I5)
2 FORMAT (16I5)
3 FORMAT (5F14.9)
4 FORMAT (4F10.3)
5 FORMAT (6F10.4)
6 FORMAT (3F10.4)
7 FORMAT ('I1')
8 FORMAT (/ 5X, 'SECTION', '13', 'RAILS=', 12.2,
* 'MAI=', 13, 'T40', 'STAGE=', 9A, 'AREA=', 11X,
1 'HYD. RAD.', '3A', 'SURF. WIDTH', '705, 93 (1-H)'
9 FORMAT (20X, 'NO. OF SECTIONS=', '13', '///
10 FORMAT (/ '20X, 'NO. OF SUBCHANNELS =', '15', '///
11 FORMAT (10X, 'WHOLE SECTION', '///', '9X, 'STAGE', '9X, 'AREA', '11X, 'HYD. RA
*D=', '5X, 'WAT. SURF. WIDTH', '///
12 FORMAT (/ '5X, 'DHO (L,1) ', '///
13 FORMAT (/ '5X, 'D50 (L)', '///
14 FORMAT (15A4)
15 FORMAT ('I1', '///', '10X, '70 (**), '///', '10X, '***', '96X, **', '///', '10X, '***', '4X,
8 '15A4, '96X, **', '///', '10X, '70 (**), '///', '10X, '70 (**), '///
16 FORMAT ((T37, 4 (F10.3, 5X)))
17 FORMAT (/ '15X, 'D50 =', 'F6.4, '4X, 'HL')
18 FORMAT (20X, 'REACH LENGTHS (F1.2) ', '///
19 FORMAT (25X, 'REACH', '13', '///', '2X, 'F10.2', '///
20 FORMAT ('///
21 FORMAT (20X, 'DISCHARGE (CFS) ', '///
22 FORMAT (30X, '8F10.0')
FORMAT (/5X, 'ITUB(L)', /)
FORMAT (/5X, 'B52(L)', /)
FORMAT (/5X, 'ALP(A), F0.3, ZA, 'BETA=A, F0.3, 2X, 'C21=, F0.3, 2X, #
    2X, 'C22=, F0.3, 2X, 'DEVA=, F0.3, 2X, 'CARE=A, F0.3, 2X, /)
FORMAT (15X, 'GET10.1, /)
FORMAT (3F10.4)
FORMAT (/, 'O(NL)')
FORMAT (/, 5X, 'I2, 2X, NTRB=1, I1, 2X, NT=1, I5, 2X, MAXMA='
    I1, 2X, 'IDEB=1, I1, 2X, 'NCYP=1, I5, 2X, 'IDOP=1, I5)
FORMAT (/5X, 'INF'=1, I1, 2X, !TRIB='!, I1, 2X, 'HEM='1, I1, 2X, 'HGD='1, I1, 2X, 'ITRS='1,
    I1, 2X, 'IBUG='I2, 2X, 'ICHG='I2, 2X, 'ICOP='I2, 2X, 'ITMAX='I2,
    I1, 2X, 'NRES='I2, 2X, 'LASTH='1, I5, 2X, 'IDWS='I2, 2X, 'IDM='I2, 2X, 'NEWS='I1, I5)
FORMAT (/5X, 'M(A(I))')
FORMAT (12X, '1015')
FORMAT (25X, 'SECTION... ', I3, ')
FORMAT (/, '10X, 'TIME INTERVAL='14, 2X, 'DAYS', ')
FORMAT (/, '10X, 'SEDIMENT CONCENTRATION IS GIVEN
*AT MOST UPSTREAM SECTION AS INP='1, 26X, '5(*-1)', ')
FORMAT (8F10.2)
FORMAT (8F10.7)
FORMAT (/, '5X, 'TF(1)')
FORMAT (/5X, 'VISC(SQ.FT./S)*L05')
FORMAT (/5X, 'TEMP')
FORMAT (/, '20X, 'SECTION... ', I3, '7X, 'STAGE', '7X, AREA', 9X
*HYD.RAD.')
FORMAT (/5X, 'S=1, F10.7, 4X, 'GTRK=1, F10.5, 4X, 'GTRK1=1, F4.2,
14X, 'ADJRI=1, F5.2')
FORMAT (/, '5X, 'SSC(I,X)')
FORMAT (15X, 'F10.4, 1X, F10.4, 1X, F10.4, 1X, F10.4, 1X, F10.4, 1X, F10.4,
*1X, F10.4, 1X, F10.4, 1X, F10.4)
FORMAT (/, '5X, 'DB(I)')
FORMAT (12X, '15')
FORMAT (/, '5X, 'DW(1)')
FORMAT (/, '5X, 'CDF(1), (T17, 10F10.4))
FORMAT (/, '5X, 'GT(1)')
FORMAT (/, '12, 'ALACH LENGTHS: ', (T17, 10F10.9))
FORMAT (/, '5X, 'DSO(S)')
FORMAT (/5X, 'VAV')
FORMAT (/, '5X, 'IDELT')
FORMAT (/5X, 'POSCITY':, (T17, 10F10.4))
FORMAT (/, '10X, 'NO.OF SECTIONS='1, I3, '10X, 'NO.OF SECTIONS='1,
*I2, '10X, 'NO.OF SECTIONS='1, I2, '/)
FORMAT (/, '5X, 'SP(1)')
FORMAT (/, '5X, 'STAGE(N))'
FORMAT (/, '25X, 75(* *)', '27X, 'WATER SURFACE PROFILE MEANS THE SAME AS IN THE PREVIOUS TIME PERIOD', '25X, 75(* *)', ')
FORMAT ((T53, '10F6.3))
FORMAT (/, '5X, 'DK(1)')
FORMAT (/, '5X, 'VIS=S, F10.7, 2X, 'SQ.FT./SEC')
FORMAT (15X, 'F10.6, 1X, F10.6, 1X, F10.6, 1X, F10.6, 1X, F10.6, 1X, F10.6, 1X,
* F10.6, 1X, F10.6, 1X, F10.6)
FORMAT (/, '10X, 25(* *)', '4X, 'WATER SURFACE PROFILE CALCULATIONS
* AFTER *1X, I4, 2X, 'DAYS', '4X, 25(* *)', '/)
FORMAT (/, '1X, 'PARTICLE SIZES', (T17, 10F10.4))
FORMAT (/, '10X, 'MODIFIED SECTION PROPERTIES AFTER', '1X, I4, 1X,
*DAYS', '10X, 41(*-1)', '/)
DO 9001 I=2,18
   TIME1(I)=TIME1(1)/364.
9001 CONTINUE
   TIME=TIME1(I),INT1,INT2.
   TIME1(1)=(INT1-CPU)/36400.*TIME1(1)
   CALDF(1)=CALDF(1)+1.
   WRITE(6,515)
515 FORMAT('00,6X,'**IDENTIFY CPU TIME DISTRIBUTION**')
   WRITE(6,501)
501 FORMAT('00,6X,'DARSI INC. RESPONSE,SHIELD')
   WRITE(6,502)
502 FORMAT('00,6X,'DATRAS INC. SHIELD')
   WRITE(6,503)
503 FORMAT('00,6X,'DAWAT INC. SECPRO,RESIST1(INC. RESPONSE,SHIELD),'*,
   *THASP(INC. SHIELD')
   WRITE(6,504)
504 FORMAT('00,6X,'DAYSO INC. VSORT')
   WRITE(6,505)
505 FORMAT('00,6X,'DAKRON INC. SHIELD')
   WRITE(6,506)
506 FORMAT('00,6X,'DASLO INC.SHIELD,HYSORT,ARMOR(SHIELD)')
   WRITE(6,507)
507 FORMAT('00,18X,'DARSI',6X,SHIELD',6X,'DATRAS',6X,'DAWAT',6X,'DAYSO',6X,'DASTT',6X,"RESPONSE",6X,"ARMOR")
   WRITE(6,508) (TIME1(I),I=1,6)
508 FORMAT('00,6X,'CPUT',6X,(2X,F10.5))
   WRITE(6,509) (CALDF(I),I=1,6)
509 FORMAT('00,6X,'CALDF',6X,(2X,F10.5))
   WRITE(6,510)
510 FORMAT('00,18X,'DARSI',6X,'SHIELD',6X,'DATRAS',6X,'DAWAT',6X,'DAYSO',6X,'DASTT',6X,"RESPONSE",6X,"ARMOR")
   WRITE(6,511) (TIME1(I),I=7,12)
511 FORMAT('00,6X,'CPUT',6X,(2X,F10.5))
   WRITE(6,512) (CALDF(I),I=7,12)
512 FORMAT('00,6X,'CALDF',6X,(2X,F10.5))
   WRITE(6,513)
   WRITE(6,514) (TIME1(I),I=15,18)
514 FORMAT('00,6X,'CPUT',6X,(2X,F10.5))
   WRITE(6,515) (CALDF(I),I=15,18)
515 FORMAT('00,6X,'CALDF',6X,(2X,F10.5))
   WRITE(6,517) IT,TIMET
517 FORMAT('00,6X,"TIME STEP =",1X,15,6X,"CPU/TIME STEP =",
   *F10.5)
   WRITE(6,518) IT,TIME,TIME1
518 FORMAT('00,6X,"TOTAL NO. OF ITERATION =",1X,15,"CPU/ITER =",
   *F10.5)
   WRITE(6,519) TIME1
519 FORMAT('00,6X,"CPU/ITER/POINT/SIZE FRACTION =",F10.5)
RETURN
END
SUBROUTINE OUTPUT

SUBROUTINE OUTPUT(KYR,LAY, VOLUME, CUMVOL, DMS, VAV, E, SE, CTO,
1 STAGE1, TALWG, QSUPS, LLM, VOLOUT, RMILE, Q, STAGE,
2 DH, ACF, ARSE, PT, QTR, QSTR)

DIMENSION DMS(N), VAV(N), E(N), SE(N), CTO(N), STAGE1(N, NMAXA),
1 TALWG(N), LLM(N), VOLOUT(N), RMILE(N), Q(N), STAGE(N),
2 ACF(N), DH(N), PT(NTRI), QTR(NTRI), QSTR(NTRI)
COMMON/INS/N,N,1,NT, N1, LAMDA, N1, MEMO, N1, IGR, N1P1, NTONA, NOES,
1 NT365, NT3652, NRE, NBL, IBED, MAXBED, NBED, NT1, IREJ, NTI
COMMON/SCALA/INDA, I, IDLT, VIS, ITIME, GAMA, IFLAG, IT, IINDSS, IFLAG1,
1 IILIM, MM1, IEQ, IRES, ISLD, ALFA, BETA, C21, C22, FFIX, IFHR, IPRINT,
2 IINF, STR, CASM, CPPHUR, IBUG, ICHE, ICOFF, VISLOG, THETA
COMMON/SHAK/IBED, QRA, IRES

PRINT OUT RESULTS

YR=ITIME/365.0
NLINE=27
N5=N/NLINE+1
N6=1
N7=N6+NLINE-1
IF(N7.GT.N) N7=N
IDAY=ITIME-KYR*360
IF((ITIME/360)*360.EQ.1ITIME) KYR=KYR+1

WRITE TIME HEADING ON RESULTS FILE

IF(NRES.GT.0) WRITE(NRES) 1T, ITIME, IDAY, KYR, VOLUME, CUMVOL
DO 811 I=1, N5
IF (IPRINT.NE.0) WRITE(6, 70) ITIME, YR
IF (IPRINT.NE.0) WRITE(6, 71)
DO 810 I=N6, N7
811 I=I+1-11
DMM=DES(I)*304.8
DEP=Q(I)/(E(I)*VAV(I))
FRR=3.0*32.2*DEP*SE(I)/(VAV(I)**2)
CBAR=C10(I)*2.65E6
DEGR=STAGE1(I, 1)-TALWG(I)
IF(IPRINT.NE.0.AND.1.EQ.0) CBA=CSTK
IF (I.EQ.N) GO TO 809
IF(IPRINT.NE.0.AND.LLM(I).EQ.0) WRITE(6, 75) VOLOUT(I)
IF(IPRINT.NE.0.AND.LLM(I).EQ.1) WRITE(6, 76) VOLOUT(I)
IF(IPRINT.NE.0) WRITE(6, 74)
1 I, RMILE(I), B(I), STAGE(I), Q(I), DEP, VAW(I), SE(I),
2 SE(I), DEGR, DH(I), TALWG(I), DM, ACF(I), CBAR, FRR
IF (NRES.LE.0) GO TO 810

WRITE CURRENT VALUES ON RESULTS FILE

WRITE (NRES) B(I), STAGE(I), Q(I), DEP, VAW(I), SE(I)
1 DEGR, DH(I), TALWG(I), DM, ACF(I), CBAR, FRR

88
WRITE (4RES) (PT(I,K), K=1,N1)
810 CONTINUE
N6=N7+1
N7=N6+NLINE-1
IF (N7.GT.N) N7=N
IF (N6.GT.N7) GO TO 105
811 CONTINUE
105 IF (NRES.GT.0) WRITE (NRES) (WIR(I), I=1,NTRIB),
1 (QSTR(I), I=1,NTRIB)
70 FORMAT(*'WATER SURFACE AND BED PROFILE AFTER', 1X, 15, 2X,
* 'DAYS', 1X, ('F', 6.2, 1A, 'YEARS'), ',/', 3X, 6Z(':-')/)
71 FORMAT ('1X, 2X, 'AM', 2X, 'PT', 2X, 'STAGE', 2X, 'Q (CFS)', 2X,
* 'DAP (PT)', 2X, 'YP', 2X, 'T', 2X, 'DI (PT)', 2X, 'DR (PT)', 2X,
* 'D50 (MM)', 2X, 'ACP', 2X, 'ACEP (PT)', 2X, 'CBAR (PMM)', 2X,
* '/', 1X, 130 ('H:-'))
72 FORMAT (1X, I3, 1A, F5.1, 2X, F5.0, 2X,
75 FORMAT (102X, '12.4')
76 FORMAT (102X, '12.4', 1X, '=')
RETURN
C#### FIRST CARD OF SUBROUTINE RARSI (DARESI) #######

SUBROUTINE RARSI

CTHIS SUBROUTINE CALCULATES SEDIMENT DISCHARGE AND FRICTION FACTOR
USING THE TOTAL LOAD TRANSPORT MODEL (TLTM) DEVELOPED AT IHR

SUBROUTINE DARESI (I, DAS, D, XAM, R, CTO, SF, ACP, CIN, B)

C DIMENSION 2(N), DAS(N), D(N), XAM(N), R(N), CTO(N), SF(N),
1 ACP(N), CIN(N), B(N)
COMMON/DIMS/N, 1, NT, 1, NAM, NMEMO, XM, IGN, NIF1, NT1, 1, NOUN, NORD,
1 NT365, NT365, NT365, NT365, NT365, NT365, NT365, NT365, NT365, NT365,
NT365, NT365, NT365, NT365, NT365, NT365, NT365, NT365, NT365, NT365,
1 ITLINIT, N1, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU,
1 IMU, ICHD, ICHD, ICHD, ICHD, ICHD, ICHD, ICHD, ICHD, ICHD, ICHD, ICHD,
1 IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU,
1 IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU,
1 IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU,
1 IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU,
1 IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU,
1 IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU,
1 IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU,
1 IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU,
1 IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU,
1 IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU,
1 IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU,
1 IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU,
1 IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU,
1 IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU,
1 IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU,
1 IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU,
1 IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU,
1 IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU,
1 IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU,
1 IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU, IMU,
CON3=32.2*1.55
CON5=32.2/1000.0
CON4=8.0*32.2
CALL RESPUN(Q, XAREA, R, DSM)
    TIME=TSIMPS(INT1, INT2)
    TIME1(I)=(INT1-CPU)/100.*TML1(7)
    CALLF(7)=CALLF(7)+1.
RETURN
**************
ENTRY RES1
**************

EVALUATE ERROR AT FIRST SLOPE ESTIMATE

    TIME=TSIMPS(INT1, INT2)
    CPU=INT1
CALL RESPUN
S1L=SF(I)*1000.
S1=S1L
CALL RESPUN(S1L, T1L)
ITER=0
IF (ABS(T1L).LE.ERR) GO TO 152
S1R=S1L
FCHG=0.1

FIND SECOND SLOPE ESTIMATE WHICH REVERSES SIGN OF ERROR

20 S1R=S1R+(1.*FCHG*SIGN(1., T1L))
IF (S1R.GT.0.) GO TO 24
S1=0.001
GO TO 152
24 ITER=ITER+1
FCHG=MIN1(FCHG*2.0, 3.0)
IF (ITER.GT.9) STOP 1940
CALL RESPUN(S1R, T1R)
IF (T1R*T1L).GE.ERR) OR (ABS(T1)/V1.LE.ERR) GO TO 150

BEGIN REGULA FALSI ITERATION

30 DO 100 ITER=1, 60
    S1=AMAX1(1.E-9, (S1L-S1R-S1R*T1L)/(T1R-T1L))
    IF (S1.LT.S1L.OR.S1L.EQ.S1R) GO TO 150
    CALL RESPUN(S1, T1)
    IF (ABS(T1).LE.ERR) OR (ABS(T1)/V1.LE.ERR) GO TO 150

KEEP RIGHT OR LEFT SUBINTERVAL

    IF (T1*T1L.LT.0.) GO TO 30
    S1L=S1
    T1L=T1
    GO TO 100
30 S1R=S1
T1R=T1
100 CONTINUE
ITER=60

CONVERGENT ITERATION
C 150 IF (ITER.LE.10) GO TO 152
CALL ERRNO (i, IT, ITIM, AABS, T1/V1)
WRITE (6, EATS)
IF (ITER.GE.60) STOP

C 152 SF (I) = S1/1000.0
PF = CON4*DI*S1/(1000.0*V1**2)
Pf = PF

C 154 AA = 0.0
IC = 0
IF (IT.EQ.0) IC = IN 1-1
IF (IT.GT.0) IC = MIN(IP)
DO 153 IA = 1, V1, S1
153 AA = AA + D (IA)
DA = AA/FLQAI (IC)
IF (IPRINT.EQ.1) WRITE (6, 25) 1, P1, V1, S1, PF, CPN

C 23 CRI = 2.0*ALOG10 (2.0*DI/DO) + 1.14
IF (IEU.EQ.1) WRITE (0, 23) 9 (IA), DA, CRI, DI, S1, DMS (I)
FORMAT (5X, 'D1** E14.7', 2X, 'AA=' , X14.5, 2X, 'CRI=' , X14.7, 2X,
0.1**D1=' , X14.6, 2X, 'S1=' , X14.6, 2X, 'D50=' , X14.6)
PPCW = 1.0/(CRI)**2
IF (IPRINT.EQ.1) WRITE (6, 22) PF, PPCW

C 23 COMPUTE COMPOSITE FRICTION FACTOR (EQ. 5 IHR 250)

C 24 PF = C21*(1.-ACF (I)) *PF+C22*ACF (I)*PPCW
B (I) = D1
SF (I) = PF*V1**2/(CON4*DI)
S1 = SF (I) *1000.0

C 25 UPDATE SEDIMENT DISCHARGE FOR COMPOSITE FRICTION FACTOR

C 156 A4 = ALOG10 (S1)
UST = SQRT (32.2*U1*S1 /1000.0)
Ao = ALOG10 (UST/W)
RS = UST*DMS (I)/VIS
CALL SHIELD
USC = SQRT (SHM)*VAR1
TEMP = ANAX1 (0.01, UST-USC)
A17 = ALOG10 (1.0/ (VAR1)
QS = 10.0**4*(RKK + A12 *RKK + A12 *A17 *RKK3 +
0.3 +A17 * .2939)
C = CTO (I) = QS*SQRT(CON3*DMS (I)**3)/(V1*D1)
IF (IT.EQ.0) CIN (I) = CTO (I)
CTO (I) = (1.-ALPA*ACF (I)) *CTO (I)
155 CFPN = CTO (I) *2.5526
IF (IT.EQ.0) GO TO 12
12 IF (IPRINT.EQ.1) WRITE (6, 10) 1, D1, V1, V2, SF (I), PF, CFPN, ITZ, ACF (I)
10 FORMAT (/ (5X, 'D1=', 13, '2X, 'D1=', 13, '2X, 'V1=', 13, '2X, 'V2=', 13, '2X,
0.2X, 'SF=', 13, '2X, 'PF=', 13, '2X, 'CFPN=', 13, '2X, 'ITZ=', 1, '2X,
0.2X, 'ACF=', 13, '2X, 'IT=', 13, '2X, 'IT=', 13, '2X/)
**PSEUDOCODE FOR SUBROUTINE RESFUN**

```plaintext
SUBROUTINE RESFUN(Q, XAREA, N, DAS)

C
C TIME = TSTMS (INT1, INT2)
C CPU = INT1

C COMMON / DIMS / N, N1, NI, M1, MAAX, MA, MEMO, NX, IGR, N1P1, NTONA, ROBS,
1 NT3651, NT3652, NTRIB, NANK, NISD, NASED, NSED, NTF, NT1, IDREJ, NTY
C COMMON / SCALR / INDEX, 1, IDELT, VIS, CTLE, GAMA, IFLAG, 1T, INDSS, IFLAG1,
1 LIMIT, IM1, IE2, IRES, ISAD, IPA, OTA, CM1, CM2, FMIX, IFR, IPRINT,
2UFP, STR, CHAM, CFPNUP, IBUG, ICBS, ICOFF, VISLOG
C COMMON / WATRES / FCW, CEPH, FCW, IDEP, CTRIV, FF, CON3, CON4, CON5,
1 D1, V1, N, VAR1, A12, A3
C COMMON / COEFF / RRA1, RRA2, RRA3, RRA4, RRA5, RRA6, RRA7, RRA8, RRA9
C COMMON / ARM / NIND, QNA, IARM8
C COMMON / SHD / DS, SHP

C COMMON / CFUT / TIME(13), CALDF(13)

C
C TIME = TSTMS (INT1, INT2)
C TIME(5) = (INT1 - CPU) / 100.0 + TIME(5)
C CALDF(5) = CALDF(5) + 1.

RETURN

C
C ENTRY RESFU2

C
C
C TIME = TSTMS (INT1, INT2)
C CPU = INT1

C V1 = Q(I) / XAREA(I)
C D1 = R(I)
C F11 = 36.0 * VIS**2 / (CON3 * DMS(I)**3)
C F1 = SQRT(2.0 / 3.0 + F11) - SQRT(F11)
C VAR1 = SQRT(CON3 * DMS(I))
C W = F1 * SQRT(1.65 * 32.2 * DMS(I))
C A12 = ALOG10(V1 / VAR1)
C A3 = ALOG10(D1 / DMS(I))
C WLOG = ALOG10(W)
C DMSLOG = ALOG10(DMS(I))

C
C TIME = TSTMS (INT1, INT2)
C TIME(5) = (INT1 - CPU) / 100.0 + TIME(5)
C CALDF(5) = CALDF(5) + 1.

RETURN

C
C ENTRY RESFUN(S1, T1)

C
```
TIME=TS TEP S (INT1, INT2)
CPU=INT1

100 A4=ALOG10 (S1)
UST=SQRT (D1*S1*CONS)
A6=ALOG10 (UST/4)
A10=WLOG+DMSLOG-VISLOG
RS=UST*XAS (1)/V1S
CALL SHIELD
USC=SQRT (SHP)*VAR1
TEMP=AMAX1 (UST-USC, 0.001)
A17=ALOG10 (TEMP/VAR1)

QS COMPUTED FROM EQ. 1, ILHR 250

QS=10.0** (RKK 1+A12 *RKK2+A12 *A17 *RKK3+
* A3 *A17 * .2939)

120 A1=ALOG10 (QS)

J COMPUTED FROM EQ. 2, ILHR 250

RIGHT=10.0** (RKK5+A1 *RKK6+A1 *A6 *A10 *RKK7+
* A6 *A10 *RKK9+A3 *A4 *A6 *(RKK9))

214 V2=RIGHT*VAR1
T1=V1-V2
TIME=TS TEP S (INT1, INT2)
TIME1 (S)=TIME1 (S) + (INT1-CPU)/100.
CALDF (S)=CALDF (S)+1.
RETURN
END

FIRST CARD OF SUBROUTINE ARMOIR (DAARMO) #

SUBROUTINE ARMAIR (ACF, ACF1, PT, D, P, CDFP, PTT, YOLOUT, BB, SF,
* D50, TBED, PTA, Nwel, STAGE1, PHED, J, KAL, DARM, E1, ARM, PAC,
& TOED, ARM)

#

TIME=TS TEP S (INT1, INT2)
CPU=INT1

THIS SUBROUTINE CALCULATES ARMOIR OF BED SURFACE

IF L= INDEX VARIABLE FOR INCLUDING BED LAYER IN SEDIMENT
CALCULATIONS; 0 IF BED LAYER IS NOT INCLUDED (IMPLYING
USE OF ARMOIR PROCEDURE); 1 IF BED LAYER IS
CONSIDERED (ARMOIR PROCEDURE NOT USEd)
IARM=INDEX VARIABLE TO SPECIFY OR DETERMINE ARMOIR
SEDIMENT SIZE; IARM=0, SPECIFY; 1 FOR
DETERMINING INTERNALLY
MIND= FRACTION NUMBER FOR THE MINIMUM SEDIMENT SIZE WHICH
(CR AND COARSER FRACTIONS) FORM ARMOR COAT
QMAX=MAXIMUM WATER DISCHARGE (CPS.) FOR DETERMINING NON-MOVING
ARMOIR FRACTIONS
IQMAX=INDEX VARIABLE FOR SPECIFYING WATER DISCHARGE FOR ARMORING
CALCULATIONS; IQMAX=0 FOR USING SPECIFIED CONSTANT
QMAX IN EACH TIME STEP AND REACH; 1 FOR USING WATER
DISCHARGE FROM SPECIFIED HYDROGRAPH
KARM=INDEX VARIABLE FOR MODIFYING THE ARMORING COEFFICIENT
(CARM); 0 FOR USING SPECIFIED VALUES; 1 FOR MODIFYING
CARM USING GESSLER'S RELATION; 2 FOR USING RED-LOAD
METHOD; 3 FOR USING BOTH GESSLER'S AND BED-LOAD METHODS;
4 FOR USING DUNE-HEIGHT METHOD; 5 FOR USING BOTH
GESSLER'S AND DUNE-HEIGHT METHODS.

IACF = INDEX VARIABLE FOR SPECIFYING INITIAL ARMORING; 0 FOR NO
INITIAL ARMORING AREA; 1 IF ARMORED COAT (PARTIAL OR FULL)
EXISTS AT THE BEGINNING.

DIMENSION P(N1), Q(N1), PT(NN, N1), PTA(NN, N1), CDEP(N),
1 VOLUM(N), ACF(N), ACF1(N), DB(N), SF(N), D50(N), THELD(N),
2 PTA(NN, N1), ACF2(N), STAGE1(N, NMAX), PBED(NN, N1), XXBAR(N, N1),
DIMENSION Q(N), RA(N), DAMM(N), E11(N), AAR(N, N1), PAC(N, N1),
DIMENSION TDELD(N, N1), HRMP(N, N1),
COMMON/DK5/N, N1, NT, NT1, NMAX, NN, LMEMO, NN, K1, K1P, NTONX, NODS,
1 NT3651, NT3652, NTRI, NTRAN, ISBED, ISBED, ITBED, NIP, NT1, lDREJ, NTY,
COMMON/SCAL/L, INDA, ILGR, ITGR, ITHE, GAMMA, IFLAG, IT, INDSS, IFLAG1,
1 LIMIT, AM1, I5Q, INES, ISED, ALFA, BETA, C21, C22, FAIX, IFQ, IPRINT,
2 IUP, STR, CARR, CFRAUP, IDUG, ICRL, ICOFF
COMMON/SLOHS/1, L7
COMMON/COEFF/RK41, RK52, RKK1, RKK2, RKK4, RKK5, RKK6, RKK7, RKK8, RKK9
COMMON/ARM/ARCH, QMAX, IARMOR, IQMXX
COMMON/SBEJ/KDREJ
COMMON/ARM/1, IELR
COMMON/SNDR/NN, SHF
COMMON/CFUT/TIMED1(18), CALLF(18)

READ INDEX VAR. FOR USE OF BED LAYER.

READ (5, 1000) IELR
IF (IELR.EQ.0) WRITE (6, 41)
IF (IELR.EQ.1) WRITE (6, 42)
41 FORMAT ('/10X, 'BED LAYER IS NOT CONSIDERED (ARMORING PROCEDURE'),
1 'USED ') '/10X, 53 ('-') , /
42 FORMAT ('/10X, 'BED LAYER IS CONSIDERED (ARMORING PROCEDURE NOT'),
1 'USED ') '/10X, 53 ('-') , /
IF (IELR.EQ.1) GO TO 4000

READ ARMORING INPUT PARAMETERS DURING PREPARATORY PHASE
OF RUN.

READ (5, 1000) IARMOR, MIND, QMAX, IQMXX, IACF, KARM, C1
1000 FORMAT (215, F10.0, 315, F10.4)
DARMOR = 0.
IF (IARMOR.EQ.0) DARMOR = 0 (MIND) * 304.3
WRITE (6, 43) IARMOR, MIND, QMAX, IQMXX, DARMOR, IACF, KARM, C1
43 FORMAT ('/10X, BED ARMORING PARAMETERS: IARMOR=', I2, ' MIND=', 1
1 I3, ' QMAX=', F10.2, ' IQMAX=', 12., ' IACF=', 12., ' KARM=', 12., ' C1=', 16., '/12',
3 T10, 24 (1H-))

C IF (IACF.EQ.1) GO TO 2000
DO 1010 I1 = 1, NN
DO 1010 K1 = 1, N1
ARM (I1, K1) = 0.0
1010 CONTINUE
GO TO 2500
2000 DO 2010 I1 = 1, NN
READ (5, 101) (ARM (I1, K1), K1 = 1, N1)

94
WRITE (6, 110)
WRITE (6, 120) (ARM(I1, K1), K1 = 1, N1)
2010 CONTINUE
101 FORMAT (6F10.6)
110 FORMAT (/,'INITIAL ARMOR-COVERED AREA (FRACTION) :',/)
120 FORMAT (T15,10F8.4)
2500 DO 3500 I1=1,N1
ACF(I1)=0.0
DO 3000 K1=1,N1
ACF(I1)=ACF(I1)+ARM(I1, K1)
ARMF(I1, K1)=ARMF(I1, K1)
3000 CONTINUE
3500 CONTINUE
4000 IF (ILBLR.EQ.0) GO TO 5000
DO 4050 I1=1,N1
4050 ACF(I1)=0.0
5000 TIMES=TMSPS(INI1, INT2)
TIME1(12)=TIME1-CP0/100.0+TIME1(12)
CALDP(12)=CALDP(12)+1.
RETURN
C***************************************************************************
ENTRY ARMOR
C***************************************************************************
C
TIME=TMSPS(INI1, INT2)
CP0=INT1
IF (ILBLR.EQ.1) GO TO 300
C
105 S1=(SF(I)+SF(I+1))/2.0
100 IF (IARMOR.EQ.0) GO TO 200
C
C COMPUTE THE ARMORING SEDIMENT SIZE
C
IF (IQMAX.EQ.1) GO TO 170
A1=OMAX/((OM(1)+OM(I+1))*0.50)/((32.2*1.65*0.50(I)**3)
1 **0.50)
A1=ALOG10(A1)
A2=ALOG10(S1*1000.0)
VFD=10.0**(-0.4812+0.37610*A1+0.31060*A2)
VEL=VFD*SQRT(32.2*1.65*0.50(I))
DEP=OMAX/(OM(I)+OM(I+1))*0.50/VEL
USTAR=SQRT(32.2*DEP*S1)
170 IF (IQMAX.EQ.1) USTAR=SQRT(32.2*(MA(I)+MA(I+1))/2.0*S1)
DO 180 K1=1,N1
RS=USTAR*D(K1)/VIS
CALL SHIELD
USC=SQRT(SHP*32.2*1.65*D(K1))
IF (USC.GT.USTAR) GO TO 135
180 CONTINUE
185 MIND=K1
IF (USC.LT.USTAR) MIND=MIND+1
IF (MIND.LE.N1) DARMOR=D(MIND)*304.8
IF (MIND.GT.N1) DARMOR=2.0*D(R1)*304.8
IF (IBUG.EQ.1) WRITE (6, 10) MIND, DARMOR, IT, I1, I2, 3A
16 FORMAT (1X,'MIND=',I12,3A,'DARMOR=',F8.3,'IT=',I3,'I1=',I4, 'I2=',I4,
, 'I3=',I3)
200 CONTINUE
C
ADJUSTING SIZE DISTRIBUTION OF ARMORING FRACTIONS IN CASE OF VERTICAL VARIATION OF BED MATERIAL

BEDEL=STAGE1(I, 1)
IF (IBED.EQ.0) GO TO 410
IF (NEEL(I).LE.0) GO TO 410
IF (DARM(I).LE.0) GO TO 410
IF (L.EQ.1) CDEPP=CDEP(I)
IF (L.GT.1) CDEPP=(CDEP(I)+CDEP(I-1))/2.0
T1=STAGE1(I, 1)-THBED(I)+CDEPP
IF (T1.LE.BEDEL) GO TO 410
IF (T1.GT.BEDEL) GO TO 205
DO 205 K=MIND,N1
205 PTA(I, K)=PTT(I, K)
208 NB=NBEL(I)
DO 210 L=1, NB
210 CONTINUE
L=NB
215 DO 260 K=MIND,N1
ELL=STAGE1(I, 1)-THBED(I)*L+CDEPP
FIN=PTT(I, K)*THBED(I)+FLED(I, K, L)*(ELL-BEDEL)
L2=L-1
IF (IBUG.EQ.1) WRITE (6, 94) K, L2
94 FORMAT (5X,'K=',I3,2X,'L2=',I10)
IF (L2.EQ.0) GO TO 255
DO 245 LA=1, L2
245 PIN=PIN+FLED(I, K, LA)*THBED(I)
255 PTA(I, K)=PIN/DARM(I)
260 CONTINUE
400 CONTINUE
IF (IBUG.EQ.1) WRITE (6, 81) PTA(I, K), I, K
81 FORMAT (5X,'PTA=',F10.6,2X,'I=',I3,2X,'K=',I12)

COMPUTING FRACTION OF ARMORED AREA

IF (IBED.EQ.1) GO TO 450
410 DO 420 K=MIND,N1
420 PTA(I, K)=PTT(I, K)
450 CONTINUE

MODIFICATION OF CURVE BY BED-LOAD METHOD

CARM1=CARM
IF (KAR4.EQ.0 .OR. KARN.EQ.1) GO TO 460

CALCULATION OF TOTAL SED. DISCH.

DEP=(RA(I)+RA(I+1))/2.0
VEL=Q(I)/((BS(I)+BS(I+1))*0.50)/DEP
A3=ALOG10(DEP/D50(I))
A12=ALOG10(VEL/SQRT(32.2*1.65*D50(I)))
USTAR=SQRT(32.2*DEP*81)
RS=USTAR*D50(I)/VIS
CALL SHIELD
USC=SQRT(SHEP*32.2*1.65*D50(I))
IF (KARM.GE.4) GO TO 455
TEMP=USTAR-USC
IF (TEMP.LE.0.0) TEMP=0.001
A17=ALOG10 (TEMP/SQRT (32.2*1.65*D50 (1)))
QS=10.0**((RKA+12*RKK2+A12*A17*RKK3+A3*A17*RKK4)
QT=QS*SQRT (32.2*1.65*(D50 (1))**3))

CALCULATION OF RBD-LOAD DISCH. (BY SBTM)

A4=ALOG10 (S1*1000.0)
P11=30.0*VIS**2/(32.2*1.65*(D50 (1))**3)
P1=PQ*SQRT (2.3+P11)-SQRT (P11)
W=P1*SQRT (32.2*1.65*D50 (1))
A6=ALOG10 (USTAR/W)
A10=ALOG10 (W*D50 (1)/VIS)
CB=10.0**(-3.7518+A12*2.0279+A4*0.4595+A6*(-2.5055)+
A10*(-0.0932)+A17*0.7395)
ETAA=(D50 (1)*USTAR/USC)/DET
F=8.0*32.2*DEPS1/(VEL**2)
FN=0.4*SQRT (8.0/F)
WU=W/USTAR
CF=1.00
IF (WU.GT.1.00) CF=1.00/SQRT (WU)
QB=CB*DEPS1*VEL*(ETAA**((1./FN+1.)))*CF
QBQT=QB/QT
PQBQT=QBQT**C1
CARM1=CARM*PQBQT
GO TO 460
CONTINUE

MODIFICATION OF CARM BY DUNE-HIGHT METHOD

THETA=DEPS1/(1.65*D50 (1))
THETAC=USC**2/(32.2*1.65*D50 (1))
IF (THETA.LE.1.10) PDH=(1.10-THETA)/(1.10-THETAC)
IF (THETA.GT.1.10) PDH=(THETA-1.10)/(1.50-1.10)
IF (PDH.GT.1.00) PDH=1.00
CARM1=CARM*PDH
IF (IBG.EQ.1) WRITE (6,54) THETA,THETAC,PDH,CARM1
FORMAT (5X,'THETA=',F6.3,2X,'THETAC=',F6.3,2X,'PDH=',F6.3,2X,'
CARM1=',F6.3,2X,'I=',I4)

IF (MIND.GT.99) GO TO 525
DO 500 IA=MIND+1
IF (KARM.EQ.0.0) KARM.EQ.2 GO TO 496
IF (KARM.EQ.4) GO TO 496

MODIFICATION OF CARM BY GESSLER'S PROBABILITY CURVE

RS=USTAR*D(IA)/VIS
CALL SHIELD
USC=SQRT (SHP*32.2*1.65*D(IA))
TCTO=(USC/USTAR)**2
XX=(TCTO-1.0)/SQRT (2.0*0.57)
PGE0=0.50+0.50*ERF (XX)
CARM1=CARM*PGE0
IF (KARM.EQ.1) GO TO 496
IF (KARM.EQ.3) CARM1=CARM*PGE0*PQBQT
IF (KARL.EQ.5)  CARM = CARM * PGESS * PBU

498 IF (IBUG.EQ.1) WRITE (6, 55) TCTO, PGESS, VBTQ, PQBTQ, PBD, CAR IT, I, IA


2 *I1=', I14, 2X, 'K=', I12)

ARM(I, IA) = ARM(I, IA) * CARM * VOLOUT(I) * (1.0 - P(IA))

5  PTA(I, IA) = D(IA)

ARM(I, IA) = MAX1(ARM(I, IA), 0.0)

500 CONTINUE

525 ACP(I) = 0.0

DO 560 IA = 1, N1

IF (IA .LT. MIND) ARM(I, IA) = 0.0

ACP(I) = ACP(I) + ARM(I, IA)

560 CONTINUE

IF (IBUG.EQ.1) WRITE (6, 63) 1, ACP(I), PTA(I, MIND), PTA(I, N1),

2 VOLOUT(I)

550 IF (ACP(I) .GT. 1.00) ACP(I) = 1.00

IF (ACP(I) .LT. 0.0) ACP(I) = 0.0

IF (IT.EQ.1) GO TO 600

IF (DARK(I) .GE. 0.0 .AND. ACP(I) .EQ. 0.0) GO TO 600

C IF (ACP(I) .LE. ACPF(I) .AND. KREJ.EQ.0) ACP(I) = ACPF(I)

600 ACPF(I) = ACP(I)

IF (IBUG.EQ.1) WRITE (6, 62) ACP(I), I

82 FORMAT (5X, 'ACPF=', F14.6, 2X, 'PT (MIND) =',

5 'E14.6, 2X, 'PT(N1) =', E14.6, 2X, 'VOLOUT=', E14.6)

C 800 IF (IBLR.EQ.0) GO TO 6000

C CALCULATION OF BED-LAYER SIZE DISTRIBUTION

C S1 = (SF(I) + SF(I+1)) / 2.0

DEP = (HA(I) + HA(I+1)) / 2.0

UST = SQRT (32.2 * DEP * S1)

RS = UST * D50(I) / VIS

CALL SHIELD

USC = SQRT (SHP * 32.2 * 1.05 * D50(I))

TACL = D50(I) * UST / USC

VRM = 0

DO 810 K1 = 1, N1

T1 = TACL * PAC(I, K1) - TDELU(I, K1)

IF (T1 .LT. 0.) T1 = 0.0

VRM = VRM + T1

810 CONTINUE

DO 850 K1 = 1, N1

T1 = TACL * PAC(I, K1) - TDELU(I, K1)

IF (T1 .LT. 0.) T1 = 0.0

PAC(I, K1) = (T1 + (TACL - VRM) * PT(I, K1)) / TACL

IF (VRM .GT. TACL) PAC(I, K1) = PT(I, K1)

850 CONTINUE

6000 TIME = STAPS (INT1, INT2)

TIME1 (12) = (INT1 - CE) / 100 + TIME1 (12)

CALDP (12) = CALDP (12) + 1.

RETURN

END

################################ FIRST CARD OF SUBROUTINE WATPRO (DAWAT) ################################
SUBROUTINE WATPRO

THIS SUBROUTINE COMPUTES WATER SURFACE PROFILE, AVERAGE VELOCITY AND FRICTION SLOPE BY STANDARD STEP METHOD

DEFINITION OF VARIABLES

SF(I) = ENERGY GRADIENT AT SECTION I
CONVF,CONVR,CONVFH = CONVEYANCE FACTORS FOR LEFT, RIGHT AND MAIN SUBSECTION, RESPECTIVELY
VEL = VELOCITY FOR THE WHOLE SECTION
VELCF = VELOCITY COEFFICIENT
VH = VELOCITY HEAD
THETD1 = TOTAL HEAD OBTAINED BY ADDING VELOCITY HEAD TO STAGE
THETD2 = TOTAL HEAD OBTAINED BY ADDING FRICTION HEAD
DY = CORRECTION TO BE APPLIED TO THE ASSUMED STAGE VALUE
ITER = NO. OF ITERATIONS REQUIRED FOR BACKWARD COMPUTATIONS
OLDH2 = TEMPORARY LOCATION FOR STORING THETD2 OF PREVIOUS SECTION
CL, CR, CM, A1, A2 = INTERMEDIATE VARIABLES FOR BACKWARD CALCULATIONS

SUBROUTINE DAWATP(REACH, STAGE, Q, TALW, D, XAREA, R, B, CTO, SF, FR, 1, ACF, LOCTR, QTR, DAS)

DIMENSION REACH(NH), STAGE(N), Q(N), TALW(N), D(N), XAREA(N), R(N), B(N), CTO(N), SF(N), FR(N), DAS(N), ACF(N), 1 LOCTR(NTRIB), QTR(NTRIB)
COMMON/DIMN, N1, NT, N1, MAXX, YM, NEMO, DX, IGN, N1P1, NTONX, NOBS,
1 NT3651, NT3652, NTRIB, NBRK, IBED, MAXBED, NED, NTP, NT1, IDREJ, ATY
COMMON/SCALE/INDEX, I, IDELT, VIS, ITIME, GAMA, IFLAG, IT, INDSS, IFLAG1,
1 ILIMIT, M1, IEQ, IRES, ISLD, ALFA, ETA, C21, C22, PMIX, IPH, iP1M1,
1 Z1UF, STR, CAR, CPF, MPUP, IDOC, ICHB, ICOFF
COMMON/WATRES, FCW, CPE, FU, LREP, CPREV, PF, CON3, CON4, CON5,
1 D1, W, VAN1, A12, A3
COMMON/USBH, DQSDH, IDCUE, CTOC, CTOC, QSN, QSNP, ADJXRE
COMMON/COEFP/RKK1, RKK2, RKK3, RKK4, RKK5, RKK6, RKK7, RKK8, RKK9
COMMON/SHD/RS, SHF
COMMON/CPUTIME, TIME1(18), CALDF(18)

VELCF = 1.0
CON1 = VELCF / 2.0 / 32.2
ERR = 0.005
CON2 = 1.0 / 32.2
CON3 = 32.2 * 1.65
CON4 = 1.0 / CON2
TIME = TSTMPS(INT1, INT2)
TIME1(10) = (INT1 - CPU) / 100. * TIME1(10)
CALDF(10) = CALDF(10) + 1.

RETURN
***************
ENTRY WATPRO
***************
IF (IT.NE.1 .AND. MOD(I1, 10CPU) .NE. 0) GO TO 900
TIME=TOTAL TIME (INT1,INT2)
CPU=INT1
IF (IDCUP.NE.1) WRITE (6,2000) IT,ITIME
2000 FORMAT (/[ 1 * WATPRO ACTIVATED AT IT=*,I5, *,ITIME=*,I10)

COMPUTATION STARTS AT THE CONTROL SECTION
(MOST DOWNSTREAM ON SECTION 1)

I=1
ITER=0

TLTM ITERATION MANAGEMENT FOR SECTION *I*

CALL SECPRO TO OBTAIN SECTION PROPERTIES

CALL SECPRO
IF (IBUG.GT.0.AND.1.GT.1.AND.IPRINT.EQ.1) WRITE (6,55) I,ITEM

CALL RES1 FOR TLTM CALCULATION

CALL RES1
FR(I)=FF
XAREA(I)=R(I)*B(I)
VEL=Q(I)/XAREA(I)
VH=VEL*VEL*COM1
THEAD1=STAGE(I)+VH
IF (I.EQ.1) THEAD2=THEAD1

COMPUTATION STARTS AT THE HEAT SECTION

IF (I.EQ.1) GO TO 135
THEAD2=OLDTH +(SF(I)+SF(I-1))*0.5*REACH(I-1)
DEL=ABS (THEAD1-THEAD2)

135 IF (IBUG.GT.0.AND.IPRINT.EQ.1)
1 WRITE (6,50) I,XAREA(I),R(I),STAGE(I),Q(I),
* SF(I),VEL,THEAD1,THEAD2,ALPHA
IF (I.EQ.1) GO TO 140
T1=THEAD1-THEAD2
IF (DEL.LT.ERR) GO TO 140
IF (ITER.EQ.30) GO TO 140
A1=2.0*VH/R(I)
A2=3.0*SF(I)*REACH(I-1)/(2.0*R(I))
DY=(THEAD1-THEAD2)/(1.0-A1+A2)
IF (IBUG.GT.0.AND.IPRINT.EQ.1)
1 WRITE (6,20) I,DY,R(I),VEL,SF(I),FF,CPM
STAGE(I)=MAX1(TALWG(I)+0.1-STAGE(I)-DY)
ITER=ITER+1
GO TO 101

140 IF (IBUG.GT.0.AND.IPRINT.EQ.2)
1 WRITE (6,52) I,R(I),VEL,Q(I),SF(I),THEAD1,
* THEAD2,FFCW,FU,CPM,ITER,ACF(I),DMS(I),TALWG(I)
IF (ITER.GT.30) CALL ERROR4 (I,IT,ITIME,ITER,DEL)

BACKWATER ITERATION COMPLETED
DISTRIBUTE TOTAL SEDIMENT DISCHARGE AMONG SIZE FRACTIONS

CALL TRASF
I=I+1

INITIAL ESTIMATE OF STAGE OF NEXT U/S SECTION

IF (I.LE.N) STAGE(I)= MAX1(TALWG(I)+2.0,
1                  STAGE(I-1)+SP(I-1)*REACH(I-1))
ITER=1
OLDH2=THEAD2
IF (I.LE.N) GO TO 101

FORMAT (5X,'I=',I3,2X,'D=',F8.4,2X,'P=',F7.3,2X,'V=',F7.3,
*      'S=',F8.6,2X,'H=',F7.3,2X,'CBA=',F8.3)

FORMAT ('+',5X,'I=',I3,2X,'P=',F10.2,2X,'R=',F5.2,2X,'H=',F9.4,2X,
* 'Q=',F9.2,2X,'SP=',F10.2,2X,'H=',F5.2,2X,'n=',F9.4,2X,'H2=',F7.3,2X,
* 'P9.4=',F9.2,2X,'ALPHA=',F7.3,2X)

FORMAT (5X,'I=',I3,2X,'D=',F6.3,2X,'V=',F6.3,2X,'Q=',F8.1,
*      'S=',F6.6,2X,'H=',F8.3,2X,'H2=',F8.3,2X,'PCW=',F6.4,2X,
* 'PV=',F6.4,2X,'CBA=',F6.2,2X,'ITER=',I3,'T5=',ACF,F5.3,
1      'DMS=',F7.4,'BED EL=',F7.3,2X)

FORMAT ('+',15X,'I=',I3,2X,'ITER=',I3,2X,'Iteration ----',I3,
*      '+',2X,15('**'),2X,'I=',I3,2X,'I=',I3)

RECALCULATING SEDIMENT DISCHARGE IN CASE OF TRIBUTARIES

IF (NTRIBE.EQ.1) GO TO 800
DO 700 I2=2,NTRIBE
I4=LOCTR(I2)
QMAIN=Q(I4)-QTR(I2)
VL=QMAIN/(DHRE(I4))
SLN=FR(I4)*VL**2.0*CON2/K(I4)
A3=ALOG10(R(I4)/DMS(I4))
A4=ALOG10(SLN)
UST=SQRT(J2.2*R(I4)*SLN)
RS=UST*DMS(I4)/VL
CALL SRELD
VAR1=SQRT(CON3*DMS(I4))
USC=SQRT(SHFL*VAR1)
TEMP=MAX1(UST-USC,0.001)
A12=ALOG10(VL/TEMP)
A17=ALOG10(PFRF/VAR1)
QS=10.0**(RKK*1A12*RKK2*A12*A17*RKK3+A3*A17*RKK4)
CTO(I4)=QS*DMS(I4)*VAR1/(VL**5(I4))
IF (ITL.EQ.1) ACF(I4)=0.0
CTO(I4)=(1.0-ALPHA*ACF(I4))*CTO(I4)

700 CONTINUE
800 CONTINUE

TIME=TSTMP5(IN1,INT2)
TIME1(10)=(INT1-CPU)/100.*TIME1(10)
CALDF(10)=CALDF(10)+1.

900 RETURN

END

******** FIRST CARD OF SUBROUTINE QSUSBC ***************

--------------------------------------------------------

SUBROUTINE QSUSBC

--------------------------------------------------------

THIS SUBROUTINE CALCULATE UPSTREAM BOUNDARY CONDITION
SUBROUTINE QSUSBC (DMS, AREA, VAV, Q, FR, R, STAGE, DW, QST, ACF, REAC, P, TALWG, TALWGP, SE, CTO, ITER DT, VOLOUT)

DIMENSION DMS (N), AREA (N), VAV (N), Q (N), FR (N), R (N), STAGE (N),
1 DW (N), QST (NT + 1), ACF (N), REAC (N), P (N), TALWG (N), SE (N)
2, TALWGP (N), CTO (N), VOLOUT (N)

COMMON/SCALE/IND, I, DELT, VIS, TIME, GAMA, IFLAG, IT, INDSS, IFLAG1,
1 ILIMIT, AIM1, IE, IG, IRES, ISED, ALFA, ETA, C1, C2, FPIX, IFR, IPRINT,
2 IUP, STR, CARA, CEPADP, IBUG, ICHB, ICOPP, VISLOG, THETA

COMMON/DMS/N, N, NT, M, MAXMA, NN, LEMO, NW, IGR, N1P, MTONX, NOBS,
1 N3E55, NT3652, NT3652, NTHI, NSANK, IBED, MAXED, NSED, NTP, NT1, IDREJ, NTY

COMMON/USB/DQS, DSR, DCUP, CTOP, CTOC, QSN, QSNF, ADJRE

COMMON/COPP/RK1, RK2, RK3, RK4, RK5, RK6, RK7, RK8, RK9

COMMON/USD1/DW, STAGE

COMMON/USD/NS, SHP

NAMESLIST, USB/I, ITBC, V1, V2, VO, US, SC, SHP, QSNF, PD, PDP, DED
1, DW, ADJRE, QSN, AVDEP, TALW, DELT, QSNF, CTO, CTOC, DEDLST

COMPUTE UPSTREAM BED LEVEL BY REQUIRING
QS (TLM) = QS (IMPOSED)

BETA1 = ADJRE * 5280 / IDLT / 30 / REAC (NN)

THETA = 1 - BETA1

ITBC = 0

I = N

VAR1 = SQRT (32.17 * 1.05 * DMS (N))

DW = STAGE (N) - TALWGP (N)

720

CALL SECPR0

ITBC = ITBC + 1

D = AREA (N) / B (N)

VAV (N) = Q (N) / AREA (N)

SE (N) = FR (N) * VAV (N) * VAV (N) / (B * 32.17 * H (N))

V1 = VAV (N) * VAR1

V2 = DW (N) / DMS (N)

UST = SQRT (32.17 * DW (N) * SE (N))

RS = UST * DMS (N) / V1

CALL SHIELD

USC = SQRT (SHP) * VAR1

TEMP = MAX1 (0.001, UST - USC)

V6 = TEMP / VAR1

QSIM = MAX1 (0.0001, UST / (1.0 - ACF (N)) / B (N))

DELTZ = STAGE (N) - DW - TALWGP (N)

V8 = (QSIM - BETA1 * REACH (NN) * (1 - P (NN)) * 

VAR1 * STAGE (N) - DW - TALWGP (N)) / (IDLT * 36400 * (1.0 - ACF (N))) / VAR1 * 

DMS (N))

IF (V8 .LE. 0.) GO TO 9

PD = ALOG10 (V6) + 2.786 - RK2 * ALOG10 (V1) - RK3 * ALOG10 (V1) * 

ALOG10 (V6) - RK4 * ALOG10 (V2) * ALOG10 (V6) 

PDP = (BETA1 * REACH (NN) * (1 - P (NN)) / (IDLT * 36400 * (1.0 - ACF (N)) 

VAR1 * DMS (N))

V8 = V8 + RK2 * B (N) / AREA (N) + RK3 * UST * B (N) / (VAR1 * V6 * AREA (N)) *

ALOG10 (V1) + RK4 * B (N) * ALOG10 (V6) / AREA (N) + RK5 * UST * B (N) / 

VAR1 * V6 * AREA (N) + ALOG10 (V2) - RK4 * ALOG10 (V6) / DW (N)) / 2.303

DELD = -PD / PDP

V8 = V8

DWP = DW

BNP = B (N)
DWNP = DW(N)
GO TO 12
10
DELD=0.
TALWG(N) = STAGE(N) - DWNP
B(N) = BNP
DW(N) = DWNP
VB = VBP
GO TO 13
12
DW = DW+DELD
TALWG(N) = STAGE(N) - DW
13
TALW = TALWG(N)
ADEX = DW(N)
IF (ABS (DELD) * GT. 0.001 * DW(S) AND. ITBC.LT. 50) GO TO 720
IF (IT.LE.1) QSN = QSN(N) / (1. - ACF(N)) / B(N)
IF (IT.LE.1) CTOC = QSN
IF (ITERDT.NE.1) GO TO 10
CTOP = CTOC
QSNP = QSN
10
QSN = VARI*DMS(N) * VS
CTOC = CTOC(N) - (Q(N-1) * B(N-1) / B(N))
DELDST = BETA * (QSN - CTOC) * THETA1 * (QSNP - CTOP)
DELDST = DEOLT*6400. + DELDST / ((1. - P(N)) * (1. - BETA1) * REACH(NN))
VOLOUT(N-1) = -DELDST
RETURN
****** FIRST CARD OF SUBROUTINE TRASF (DATRAS) ******
SUBROUTINE TRASF
--------------------------------------------------
THIS SUBROUTINE CALCULATES SEDIMENT DISCHARGE BY SIZE
FRACTION BY IHR METHOD (EQ.20.IHR 250)
-------------------------------------------------------------------
SUBROUTINE DATRAS(D50,D,P,SSC,R,CP0,SP,PS,PDN,PAC)
--------------------------------------------------
TIME = TSTMR (INT1.INT2)
CPU = INT1
DIMENSION D50(N), D(N), P(N), SSC(N), R(N), CTO(N),
1 SP(N), PS(N), PDN(N), PAC(N)
COMMON/ D(M, N, 1, NT, M1, MAXA, NN, MEMO, TX, IGR, NP1, NTONX, NOBS,
1 NT3651, NT3652, NTR, MBANK, IED, MAXBED, NBD, NTF, NT1, IDREJ, NTY
COMMON/SCALE/INDEX, I, IDELT, VIS, ITIME, GAM, IFLAG, IT, INDS1, IFLG1,
1 ILIMIT, HM1, IQ, IH, IED, ALPA, BETA, C21, C22, FMIX, IPR, IPRINT,
21UP, STR, CAR, CPPMUP, IDUG, ICHB, ICOPF
COMMON/ WATRES/FCW, CPPH, PU, HREP, CPREV, FP
COMMON/ BANK/ISSE, QMIN, IUPS
COMMON/ ARM1/ IBLR
COMMON/ SHDF, RS, SHE
COMMON/ CPUT/TIME1(18), CALDF(18)
TIME = TSTMR (INT1, INT2)
TIME1(9) = (INT1-CPU)/100.+TIME1(9)
CALDF(9) = CALDF(9) + 1.
RETURN
ENTRY TRASF
TIME = TSTAPS (INT1, INT2)
CPU = INT1
IF (I.EQ.N) GO TO 100
SS = 0
X = 0.0316 * (k (I) / D50 (I))**0.50
DO 160 K = 1, N1
IF (IT.EQ.0) PAC (I, K) = PT (I, K)
UST = SQRT (32.2 * K (I) * SF (I))
RS = UST * D (K) / V15
CALL SHIELD
USC = SQRT (SHF * 32.2 * 1.65 * D (K))
PROB = 1.00
IF (UST.LT.USC) PROB = 0.00
IF (IBLR.EQ.0) SS = SS + PT (I, K) * ((D50 (I) / D (K))**X) * PROB
IF (IBLR.EQ.1) SS = SS + PAC (I, K) * ((D50 (I) / D (K))**X) * PROB
CONTINUE
N2 = N1 - 1
DO 200 K = 1, N1
UST = SQRT (32.2 * K (I) * SF (I))
RS = UST * D (K) / V15
CALL SHIELD
USC = SQRT (SHF * 32.2 * 1.65 * D (K))
PROB = 1.00
IF (UST.LT.USC) PROB = 0.00
PSI = 0.0
IF (SS .NE. 0.0) PSI = PT (I, K) * ((D50 (I) / D (K))**X) / SS * PROB
IF (SS .NE. 0.0 .AND. IBLR .EQ. 1) PSI = PAC (I, K) * ((D50 (I) / D (K))**X) / SS * PROB
PQS (I, K) = PSI
SSC (I, K) = COS (I) * PSI
IF (IBUG .EQ. 1) WRITE (6, 37) I, K, SSC (I, K)
37 FORMAT ('5x, '5x, '12', '8.16E')
CONTINUE
S2 = 0
DO 250 K = 1, N1
S2 = S2 + PQS (I, K) * (D (K) / D50 (I))**X
DO 260 K = 1, N1
PDN (I, K) = PQS (I, K) * (D (K) / D50 (I))**X / S2
IF (IBLR .EQ. 0) PDN (I, K) = PT (I, K)
IF (IBLR .EQ. 1) PDN (I, K) = PAC (I, K)
CONTINUE
100 IF (INDSS.EQ.1 .AND. IUPS .EQ. 0) CTO (I) = CPPHVI / (2.65 * 1000000.0)
     TIME = TSTAPS (INT1, INT2)
     TIMEL (9) = (INT1 - CPU) / 100 + TIME (9)
     CALDF (9) = CALDF (9) + 1.
RETURN
END

C### FIRST CARD OF ROUTINE SLLOAD (DASLOA) ###

---

SUBROUTINE SLLOAD

---

THIS SUBROUTINE APPLIES SEDIMENT CONTINUITY EQUATION AND UENT
CALCULATES DEPTHS OF DECK / AGGR.
DEFINITION OF VARIABLES

SSC(I,K) = SUSPENDED SEDIMENT CONCENTRATION AT SECTION I, SEGMENT J, SED. FRACTION K
EE(I) = WIDTH OF SECTION I, SEGMENT J
W(K) = FALL VELOCITY OF SEDIMENT PARTICLE OF FRACTION K
EREL(I,K) = EROSION VELOCITY AT SECTION I, SEG, J, SED, FR. K
CB(I,K) = SEDIMENT CONCENTRATION IN BED LAYER AT SECTION I, SEGMENT J, SED. FRACTION K
EZ(I,K) = TRANVERSE DIFFUSION COEFFICIENT FOR SEDIMENT IN SECTION I, SEGMENT J, SED. FR. K
N = NO. OF SECTIONS (IN LONGITUDINAL DIRECTION)
NN = NO. OF REACHES = N-1
M1 = NO. OF SEGMENTS
N1 = NO. OF SEDIMENT SIZE FRACTIONS
REACH(I) = LENGTH OF REACH BETWEEN SECTIONS I AND I+1
MANG(I) = MANNING'S COEFFICIENT AT SECTION I, SEGMENT J
VAV(I) = AVERAGE VELOCITY AT SECTION I, SEGMENT J
DELT=TIME INTERVAL (DAYS)
P(K) = POROSITY FOR SEDIMENT FRACTION K
SI=PARAMETER FOR TRANSVERSE SEDIMENT TRANSFER
CC, CD, TEST=INTERMEDIATE VARIABLES FOR SUSP. SEDIMENT CALCULATIONS
KA(I,K) = INDEX VARIABLE TO INDICATE WHETHER LIMITING SCOURING DEPTH IS REACHED AT SECTION I, SEGMENT J, SED. FR. K (=0, LIMITING DEPTH NOT REACHED; =1, LIMITING DEPTH REACHED)

NOTE: RA AND BB ARE PSEUDONYMS FOR R AND B

SUBROUTINE DASIOA (REACH, P, D, PT, TT, SSC, CTO, SF, ACF, KA, EE, 1 1
1, DELDS, TDDEL, CDEP, KA, VOLOUT, DELT, PQS, PDN, CDEP1, 2
2, TB, ACF1, LOC1, QS1, STRIB, TDELT, LOCBER, BERS, PBANK, Q, PTA, 3
3, D50, TBED, NBE, FBD, DARM, SL1, VAV, QSDF)

TIME=TSTEPS (INT1, INT2)
CPU=INT1

DIMENSION REACH(NN), P(N1), D(N1), PT(NN,N1), PT(TN,N1), 1
1 SSC(NN,N1), CTO(N1), SF(N1), ACF(N1), RA(N1), EE(N1), 2
2 DW(N1), DELDS(N1, N1), TDDEL(N1, N1), CDEP(N1), 3
3 KA(N1, N1), VOLOUT(N1), DELT(N1, N1), PQS(N1, N1), 4
4 PDN(N1, N1), CDEP1(N1), TB(N1, N1), ACF1(N1), QSTR(NTRIB), 5
5 PSTRIB(NTRIB, N1), TOULT(NTRIB, N1), LOCTHR(NTRIB), 6
6 LOCBER(NBANK), BERS(NBANK), PBANK(NBANK, N1), Q(N1), 7
7 PTA(NN, N1), D50(NN), TBED(NN), NBE(NN), 8
8 PBED(NN, N1, MALED), DARM(N1), SL1(N1), VAV(N1), QSDF(NN)
COMMON/DINS, N1, T, M1, AAKA, NA, MEMO, NZ, IG, N1F1, N1ONX, NOES, 1
1 NT3651, NT3652, NTRIB, NBE, MALED, NBED, NTF, NT1, IDREJ, NTY 2
2 COMMON/SCALE/INDEX, N, IC, N1, VIS, LTIME, GAMA, IFLAG, IT, INOS, IFLAG1, 3
3 ILINIT, NA1, IEQ, IRES, ISED, ALFA, BETA, CZ1, CZ2, PMX, IFR, IPRO, 4
4 IUP, STR, CNRM, CPPAWP, BUC, LCHS, IOFF, VISLOG, SDAE, 5
5 COMMON/WRITPS/WFWC, CEPN, PR, INEC, CFWV, PP 6
6 COMMON/SLODES/I, K 7
7 COMMON/COEFF/RAK, AAK2, AAK3, AAK4, AAK5, AAK6, AAK7, AAK8, AAK9 8
8 COMMON/BANK/ISED, QIN, IUP 9
9 COMMON/ARM/HIND, QMAX, IARM
COMMON/SHF/RS,SHF
COMMON/CPUT/TIM1(18),CALDF(18)

CON3=1.65*32.2
THETA1=1.0-THETA
TIME=TIMPS(IN1,INT2)
TIME1(13)=(INT1-CFU)/100.+TIME1(13)
CALDF(13)=CALDF(13)+1.
RETURN
******
ENTRY SLOAD
******
TIME=TSTMPs(IN1,INT2)
CFU=INT1

LOOP ON REACHES, U/S TO D/S

DO 1000 I1=2,N
I=I+1-I1
I5=I
I6=I+1
QSTU=Q(I+1)/((DB(I)+DB(I+1))*0.5)
DELCTO=CTO(I+1)-CTO(I)
IF (I.EQ.N-1) DELCTO=0.0-CTO(I)
QSD=THETA*DELCTO+THETA1*QSDP(I)

LOOP ON SEDIMENT SIZE FRACTIONS FOR EACH REACH

DO 800 K=1,N1
VAR1=SQRT(CON3*D(K))
599 KA(I,K)=0
WPAC=1.0

COMPUTATION OF TRANSPORT CAPACITY BY SIZE FRACTION
IN EACH REACH (INCONSISTENT FOR THETA.NE.1)

DO 200 I2=I5,I6
UST=SQRT(32.2*RA(I2)*SF(I2))
RS=UST*D(K)/VIS
CALL SHIELD
USC=SQRT(SHF)*VAR1
F11=36.0*VIS**2/(CON3*D(K)**3)
F1=SQRT((2.0/3.0+F11)-SQRT(F11))
WI =F1*VAR1
A3=ALOG10(RA(I2)/D(K))
A4=ALOG10(SF(I2)*1000.0)
A12=ALOG10(WAV(I2)/VAR1)
TEMP=AMAX1(UST-USC,0.001)
A17=ALOG10(TEMP/VAR1)
QSK=10.0**(AKK1+A12*AKK2+A17*AKK3+A3*AKK4)
IF (INDSS.EQ.1.AND.I2.EQ.N) QSK=CPPMUF/2.0586
*PCTRL (I,K)
IF (I2.EQ.15) QSIK=QSK
200 CONTINUE

QSK= U/S LOAD FRACTION K
QSIK= D/S LOAD FRACTION K
QSD= GLOBAL DEFICIT, REACH 1 ( +VE FOR AGGR. )
WPAC= 1 FOR GLOBAL DEGR, ,K DEGR
IF(QSD.GT.0.0) WFAC=1.00-QS1K/QS K
IF(QSD.GT.0.0.AND.WFAC.LT.0.0) WFAC=0.0
IF(QSD.LT.0.0.AND.QS1K.LT.QS K) WFAC=0.0
IF(ABS(QSD).LE.0.377E-6) WFAC=0

COMPUTATION OF DEGRADATION OR DEPOSITION IN
REACH I FOR FRACTION K, ADJUSTED FOR INCONSISTENCY
WITH GLOBAL TRENDS USING WFAC

IF(QSD.LE.0.0) DELDS(I,K)=QSD*PCS(I,K)*QST0/((1.-P(K))*REACH(I))
1*IDELT*86400.0*WFAC
IF(QSD.GT.0.0) DELDS(I,K)=QSD*PDS(I,K)*QSTU/((1.0-P(K))
*REACH(I))*IDELT*86400.0*WFAC

TDELW(I,K)=-DELDS(I,K)

ADJUSTMENT FOR TRIBUTARY SEDIMENT INFLOWS (INCONSISTENT FOR THEMAKE.1)

DO 700 I2=1,NTRIB
I4=LOCTR(I2)-1
IF(I4.NE.1) GO TO 700
IF(INDSS.EQ.0.AND.I2.EQ.1) QSTN(I2)=CTO(I)*Q(K)
TDELTR(I2,K)=QSTN(I2)*IDELT*86400.0/((BB(I4)+BB(I4+1))
1/2.0*REACH(I4))*TRIB(I2,K)/(1.0-P(K))
TDELW(I4,K)=TDELW(I4,K)-TDELTR(I2,K)
700 CONTINUE

ADJUSTMENT FOR BANK EROSION

IF(NBANK.EQ.0) GO TO 800
WF=0.0
I4=LOCBER(I3)
IF(I4.NE.1) GO TO 725
EROS=BEROS(I3)*IDELT*REACH(14)/5280.0/(1.0-P(K))/(BB(14)
1+BB(I4+1))*0.5*REACH(I4))*WF
IF(IBED.EQ.0) PBI(N(K)=ETI(I4,K)
TDELW(I4,K)=TDELW(I4,K)-EROS*PBI(N(K)
725 CONTINUE

800 CONTINUE

CHECK CONTINUITY IN CASE OF TRIBUTARIES OR BANK EROSION

IF(NTRIB.EQ.1.AND.NBANK.EQ.0) GO TO 790
IF(QSD.EQ.0.0) GO TO 790
DO 740 I2=1,NTRIB
I3=LOCTR(I2)-1
IF(I3.EQ.1) GO TO 746
740 CONTINUE
742 DO 745 I2=1,NBANK
I3=LOCBER(I2)
IF(I3.EQ.1) GO TO 746
745 CONTINUE
GO TO 790
QSDI=0.
DO 747 L=1,N1
747 QSDI=QSDI+TDELD(I,L)
QSDO=QSDI
DO 765 L=1,N1
IF (TDELD(I,L).GE.0.0) GO TO 765
IF (PQS(I,L).NE.0.0) GO TO 765
IB=MIND
IF (IT.EQ.1) IB=N1-1
IF (L.GE.IB) GO TO 765
QSDI=QSDI-ABS(TDELD(I,L))
TDELD(I,L)=0.0
765 CONTINUE
DO 775 L=1,N1
IF (TDELD(I,L).NE.0.0) TDELD(I,L)=TDELD(I,L)+
& (QSDI-QSDO)*PQS(I,L)
775 CONTINUE

CALL HYSORT TO UPDATE MIXED LAYER SIZE
DISTRIBUTION FOR REACH I

CALL HYSORT

CALCULATE TOTAL DEGRADATION IN REACH I

TOTAL=0
DO 550 K=1,N1
550 TOTAL=TOTAL+TDELD(I,K)
IF (ABS(TOTAL).LE.0.001) TOTAL=0.0
AB=TOTAL/DW(I)
IF (IT.EQ.1) GO TO 570
CDEP(I)=CDEP1(I)+TOTAL
DARM(I)=CDEP(I)
GO TO 571
570 CDEP(I)=TOTAL
DARM(I)=TOTAL

CALL ARMOR TO UPDATE ARMORING FACTOR FOR REACH I

571 CALL ARMOR
900 CONTINUE
1000 CONTINUE

PRINT OUT OF RESULTS

1005 IF (IBUG.EQ.0.OR.IPRINT.EQ.0) GO TO 1006
WRITE(6,99) IFLAG
WRITE(6,98) ITIME
1006 DO 500 I=1,N
I=N+1-I
IF (IBUG.EQ.1) WRITE(6,50) I,J
DO 500 K=1,N1
PCENT= PT(I,K) * 100.0
IF (DELT(I,K).EQ.0) DELT(I,K)=900.0
500 IF (ISED.EQ.1.AND.IBUG.EQ.1) WRITE(6,65) K,TDELD(I,K),PCENT
501 CONTINUE
IF (IBUG.EQ.0.OR.IPRINT.EQ.0) GO TO 9999

108
N4=N-1
DO 502 14=1,N1
WRITE (6,86) 14, (TDLDD (I4,K), K=1,N1)
WRITE (6,87) 14, (PI (I4,K), K=1,N1)
WRITE (6,88) 14, (FQS (14,K), A=1,N1)
WRITE (6,89) 14, (DELT (I4,K), K=1,N1)
502 WRITE (6,95)
86 FORMAT (10X,'I=',J3X,'DEGR/AGGR=',2X,'9P9.5')
87 FORMAT (10X,'I=',J3X,'S.D. (3X)=',2X,'9P9.5')
88 FORMAT (10X,'I=',J3X,'S.D. 3X=',2X,'9P9.5')
89 FORMAT (10X,'I=',J3X,'DELT (DAYS)=',2X,'9P9.2')
50 FORMAT ('///',10X,'REACH....',13X,'SEGMENT...',12X,'10X,28('-'),///
*10X,'SEM.FRACTION',15X,'CHANGE IN BED ELEVATION (FT.)',///
*10X,'BED LO
*AD',11X,'SUSP. LOAD',12X,'TOTAL',10X,'PERCENTAGE',///
65 FORMAT (15X,'=12',5OX,'=14.6',8X,'=8.4')
98 FORMAT ('///',10X,'=10X',9X,4X,'=4X',='=4X',='=4X',='=4X')
96 FORMAT ('///',10X,'A4=',1,FB,4)
99 FORMAT ('///',10X,'IFLAG=',12)
95 FORMAT ('')
9999 TIME=TSTEPS (INT1,INT2)
TIME1 (13) = (INT1-CEF)/100.+TIME1 (13)
CALDF (13) = CALDF (13) + 1.
RETURN
END

C********************************** FIRST CARD OF SUBROUTINE INFLOW *********

C C
C SUBROUTINE INFLOW (V, QTR, LOCTR, TEMPF, STAGE, QSUPS, QTRIB)
C TIME=TSTEPS (INT1,INT2)
C CPU=INT1
C D
C COMMON/DIM/N,N1,N1T,NT,MAXMA,MIN,REDU,NX,IGN,MP1,MTONK,NOBS,
C NT3651,NT3652,NTHE,BNK,ILDE,RAND,REDH,NTF,NTI1,NDREJ,NTY
C COMMON/SCAR/INDEX,1,IDELE,VIS,IDTIME,GA,IFLAG,IT,INDSS,IFLAG1,
C 1 ILIMIT,MMI,ILQ,TH,IESL,ALPHA,BETA,C21,C22,PFX,IPR,IPRINT,
C 2UFP,STR,CARE,CPLSUP,IBUG,ICHS,ICOF
C COMMON/CPUT/TIME1 (13),CALDF (13)
C IF (IT.EQ.0) IFLAG=0
C READ NEW INFLOWS IF NEEDED
C IF (ITIME.LT.1TD) GO TO 160
C DO 30 LT=1,NTRIB
C QTR (LT) =QTRIB (LT)
C 30 CONTINUE
C TEMPF=TFREAD
C STAGE (1) =YREAD
C QSUPS=QSREAD
C ITTIMEP=ITDAT
C READ (5,1000,END=500) ITDAT, TFREAD, (QTRIB (LT), LT=1,NTRIB), YREAD,
C 1 QSREAD
C WRITE (6,2002) ITDAT, TFREAD, QTRIB, YREAD, QSREAD

109
2002 FORMAT (T5,'TIME-DEPENDENT DATA READ FOR ITDAT=',15,';','
1   (T46,F6.1,10F8.0),/)
   GO TO 25
1000 FORMAT (14,P4.0,(19,3F5.0)/2P0.0)
C C C INTERPOLATE BETWEEN VALUES AT PREVIOUS TIME STEP AND THOSE
C MOST RECENTLY READ
C 100 FTIME=ITIME
     FINT=(FTIME-ITIMEP)/(ITDAT-ITIMEP)
     DO 125 LT=1,NRIV
     QTR(LT)=QTR(LT)+FINT*(QTRD(LT)-QTR(LT))
125 CONTINUE
     TEMPF=TEMPF+FINT*(ITREAD-TEMPF)
     STAGE(1)=STAGE(1)+FINT*(ITREAD-STAGE(1))
     QSUPS=QSUPS+FINT*(QSRD-QSUPS)
     ITIMEP=ITIME
C C C COMPUTE WATER DISCHARGES AT ALL COMPUTATIONAL POINTS BY
C ACCUMULATION OF TRIBUTARY INFLOWS
C Q(N)=QTR(1)
     LT=2
     LM=N
10     LM=LM-1
     IF (LM.EQ.0) GO TO 301
     Q(LM)=Q(LM+1)
     IF (LM.NE.LOCTR(LT)) GO TO 10
     Q(LM)=Q(LM)+QTR(LT)
     LT=LT+1
     GO TO 10
C C C PRINT MAINSTREAM DISCHARGES AT TRIBUTARY INFLOW POINTS
C 301 IF (IBUG.NE.0) WRITE (6,2000) IT,ITIME,NTRIM,
1   Q(LOCTR(K),K=1,NRIV)
2000 FORMAT (1X,'IT=',14,' ITIME=',14,' NTRIM=',11,' FLOWS:','
1   (T35,12F8.0))
C     TIME=TSTMS(INT1,INT2)
     TIME(17)=TIME(17)+(INT1-CPU)/100.
     CALDF(17)=CALDF(17)+1.
999 RETURN
900 WRITE (6,2001) ITIME, ITDAT
2001 FORMAT (/,'20(1H*),',ERROR: ITIME=',15,' 
1   * EXCEEDS LAST INFLOW DATA ITDAT=',15)
STOP
END
C###### FIRST CARD OF SUBROUTINE START #######
C C C SUBROUTINE START(STAGE,Q,STAGEI,DOMS,E,XARDA,MA,B1,B1,REACH,
1    FP,AREA,TALAG)
C C C TIME=TSTMS(INT1,INT2)
C CPU=INT1
DIMENSION STAGE(N), Q(N), STAGE1(N,MAXHA), DNS(N), B(N),
XAREA(N), EA(N), E1(N,MAXHA), L1(N,MAXHA), REACH(NN), SF(N)
, AREA(N,MAXHA), TALG(S)
COMMON/DIMS, N, I, NT, N1, MAXHA, NN, NEXO, NX, IGR, NIF1, NTONX, NOESS,
INT350, INT352, NTHIS, NBEAK, ISED, MAXBED, NBEV, NT, N1, IDREJ, INT
COMMON/SCAL/I, INDEX, I, IDEL1, VIS, ITIME, GA, IFLAG, IT, INDS, IFLAG1,
1 IILIMIT, MAXI, IEQ, IREQ, ISED, ALFA, DEL1, C21, C22, FMIX, IPR, IPRINT,
21UP, STR, CARD, CPB1P2, IBUC, ICHR, ICOFF
COMMON/CP1/TIME1(18), CALDF(18)

I=1
S1=STR
A2=ALOG10 (S1*1000.0)
I1=M A(I)
ITER=0
D50F=DMS(I)

100 QU=Q(I)/ST R
A1=ALOG10 (QU/SQRT (1.65*32.2*D50F **3))
VFD=10.0** (-0.4832+0.3701*A1+0.3106*A2)
VEL=VFD*SQRT (32.2*1.65*D50F )
ARE=Q(I)/VEL
DO 150 L=1,11
IF (AREA(I,L)-GT.ARE) GO TO 200

150 CONTINUE
200 C=(ARE-AREA(I,L-1))/(AREA(I,L)-AREA(I,L-1))
STAGE(I)=ST AGEL(I,L-1)+C*(STAGE1(I,L)-STAGE1(I,L-1))
B(I)=B1(I,L-1)*C*31(I,L)-E1(I,L-1))
BTR=B(I)
ERR=ABS (BTR1-BTR-1.0)
ITER=ITER+1
IF (ERR.LE.0.02) GO TO 500
IF (ITER.GT.20) GO TO 500
BTR=BTR1
GO TO 100

500 CONTINUE
STAGE(I)=STAGE(I)+TALWG(I)-STAGE1(I,1)
TIME=TIMEPS(INT1,INT2)
TIME1(18)=TIME1(18)+(INT1-CPU)/100.
CALDF(18)=CALDF(18)+1.
RETURN

C########### FIRST CARD OF SUBROUTINE LAELG (DADRED) ############

SUBROUTINE DADRED (IDR1, IDRL, NDR1, VDREJ, REACH, B, TALWG,
1 CDEP, AREA, H1, NREL, TBED, PT, TTT, P3ED, BARM, EL1, MA)

TIME=TIMEPS(INT1,INT2)
CPU=INT1

C THIS SUBROUTINE READS AND COMPUTES THE EFFECT OF DREDGING

DIMENSION IDR1(NT), IDAL(NT,N), NDR1(NT), VDREJ(N), REACH(NN),
READ DREDGING PARAMETERS

WRITE (6, 49)
READ (5, 10) ADRT
READ (5, 10) (NDRL (I), I = 1, NDRT)
WRITE (6, 52) NDRT
WRITE (6, 54)
WRITE (6, 15) (NDRL (I), I = 1, NDRT)
FORMAT (12I5)
FORMAT (15X, 12I5)
FORMAT (5F10.1)
49 FORMAT ('/,'10X,'INPUT VALUES FOR DREDGING :',//)
52 FORMAT ('/,'10X,'NDRL=',I3,//)
54 FORMAT ('/,'5X,'NDRL (I) ',//)

READ (5, 10) (IDRT (I), I = 1, NDRT)
WRITE (6, 56)
WRITE (6, 15) (IDRT (I), I = 1, NDRT)
WRITE (6, 58)
DO 350 I = 1, NDRT
L = NDRL (I)
READ (5, 10) (IDRL (I, I), I = 1, L)
350 WRITE (6, 15) (IDRL (I, I), I = 1, L)
56 FORMAT ('/,'5X,'IDRL (I, I) ',//)
58 FORMAT ('/,'5X,'IDRL (I, I) ',//)
400 CONTINUE

TIME = TSTEPS (INT1, INT2)
TIME (16) = (INT1 - CF0) / 100 * TIME (16)
CALDP (16) = CALDP (16) + 1.

RETURN

******* ENTRY DREDGE *******

TIME = TSTEPS (INT1, INT2)
CP0 = INT1

COMPUTES THE EFFECT OF DREDGING ON GEOMETRIC PROPERTIES

ITEST = 0
DO 810 IT1 = 1, NDRT
IT1 = IDRT (I)
IF (IT1 .EQ. IT) ITEST = ITEST + 1
IF (IT1 .EQ. IT) GO TO 815

810 CONTINUE

815 IF (ITEST .EQ. 0) GO TO 1000
KREJ = 1
L = NDRL (1)
DO 990 12 = 1, L
I = IDRL (1, 12)
READ (5, 30) VDREJ (I)
WRITE (6, 62) IT1, I, VDREJ (I)
DDREJ = VDREJ (I) / (BEACH (I) * (E (I) + E (I + 1)) / 2.0) * 27.0 * IDELT
CDEP (I) = CDEP (I) + DDREJ
DARM (I) = 0.0
MM = MA (I)

990 TALWG (I) = TALWG (I) - DDREJ

FORMAT ('/,'/10X, 15X, 'VOLUME OF DREDGING AT IT=', I4, 1 2X, 'I=', I3, 2X, 'IS:', F12.0, 2X, 'CU.YDS./DAY ',//, 10X, 70 (**), //)

C COMPUTE THE EFFECT OF DREDGING ON BED-MATERIAL SIZE DISTRIBUTION

IF (IBED .EQ. 1 .AND. NBEL (I) .NE. 0) GO TO 855
BELIN = TALWG (I) + CDEP (I)
DO 852 K = 1, N1
IF (EL1 (I) .LE. TALWG (I)) PT (I, K) = PT (I, K)
852 CONTINUE

855 IF (IBED .EQ. 0) GO TO 890
IF (NBEL (I) .EQ. 0) GO TO 890
NB = NBEL (I)
DO 860 L = 1, NB
T1 = BELIN - T * THBD (I)
IF (T1 .GT. TALWG (I)) GO TO 865
860 CONTINUE

865 CONTINUE

DO 890 K = 1, N1
IF (EL1 (I) .LE. TALWG (I)) PT (I, K) = PBed (I, K, L)
PT (I, K) = PT (I, K)

890 CONTINUE

1000 CONTINUE

TIME = TSTEPS (INT1, INT2)
TIME1 (10) = TIME1 (10) + (INT1 - CE0) / 100.
CALDF (10) = CALDF (10) + 1.

RETURN

END

C***************************************************************

------------------------------------------

SUBROUTINE SECPO

------------------------------------------

THIS SUBROUTINE COMPUTES CROSS SECTIONAL AREA, HYD. RADIUS AND WATER
SURFACE WIDTH AT SECTION 1 FOR A PARTICULAR ELEVATION

------------------------------------------

SUBROUTINE DASECP (STAGE, R, SIAGE1, 31, R1, AREA, XAREA, R3, D, TALWG)

------------------------------------------

TIME = TSTEPS (INT1, INT2)
CPU = INT1

113
DIMENSION STAGE(N),XA(N),STAGEI(N,MAXMA),HI(N,MAXMA),
  R1(N,MAXMA),AREA(N,MAXMA),XAREA(I),R(N),B(N),TALWG(N)
COMMON/LMS/NT1,NT2,NTM,NTM1,NTM2,NN,NO,SX,IGN,NIP1,NOTONX,NODES,
  NT3TB51,NT3TB52,NT3TB53,NT3TB54,NT3TB55,NT3TB56,NT3TB57,NT3TB58,
COMMON/SCALE/INDEX,1,INDEX,VIS,INDEX,TIME,2,INDEX,INDEX,INDEX,INDEX,INDEX,
  INDEX,INDEX,INDEX,INDEX,INDEX,INDEX,INDEX,INDEX,INDEX,INDEX,INDEX,
COMMON/STAGE/STAGEI,STAGE1,STAGE2,STAGE3,STAGE4,STAGE5,STAGE6,STAGE7,STAGE8,
  STAGE9,STAGE10,STAGE11,STAGE12,STAGE13,STAGE14,STAGE15,STAGE16,
  STAGE17,STAGE18,STAGE19,STAGE20,STAGE21,STAGE22,STAGE23,STAGE24,
  STAGE25,STAGE26,STAGE27,STAGE28,STAGE29,STAGE30,STAGE31,STAGE32,
  STAGE33,STAGE34,STAGE35,STAGE36,STAGE37,STAGE38,STAGE39,STAGE40,
  STAGE41,STAGE42,STAGE43,STAGE44,STAGE45,STAGE46,STAGE47,STAGE48,
  STAGE49,STAGE50,STAGE51,STAGE52,STAGE53,STAGE54,STAGE55,STAGE56,
  STAGE57,STAGE58,STAGE59,STAGE60,STAGE61,STAGE62,STAGE63,STAGE64,
  STAGE65,STAGE66,STAGE67,STAGE68,STAGE69,STAGE70,STAGE71,STAGE72,
  STAGE73,STAGE74,STAGE75,STAGE76,STAGE77,STAGE78,STAGE79,STAGE80,
  STAGE81,STAGE82,STAGE83,STAGE84,STAGE85,STAGE86,STAGE87,STAGE88,
  STAGE89,STAGE90,STAGE91,STAGE92,STAGE93,STAGE94,STAGE95,STAGE96,
  STAGE97,STAGE98,STAGE99,STAGE100,STAGE101,STAGE102,STAGE103,STAGE104,
  STAGE105,STAGE106,STAGE107,STAGE108,STAGE109,STAGE110,STAGE111,STAGE112,
  STAGE113,STAGE114,STAGE115,STAGE116,STAGE117,STAGE118,STAGE119,STAGE120,
  STAGE121,STAGE122,STAGE123,STAGE124,STAGE125,STAGE126,STAGE127,STAGE128,
  STAGE129,STAGE130,STAGE131,STAGE132,STAGE133,STAGE134,STAGE135,STAGE136,
  STAGE137,STAGE138,STAGE139,STAGE140,STAGE141,STAGE142,STAGE143,STAGE144,
  STAGE145,STAGE146,STAGE147,STAGE148,STAGE149,STAGE150,STAGE151,STAGE152,
  STAGE153,STAGE154,STAGE155,STAGE156,STAGE157,STAGE158,STAGE159,STAGE160,
  STAGE161,STAGE162,STAGE163,STAGE164,STAGE165,STAGE166,STAGE167,STAGE168,
  STAGE169,STAGE170,STAGE171,STAGE172,STAGE173,STAGE174,STAGE175,STAGE176,
  STAGE177,STAGE178,STAGE179,STAGE180,STAGE181,STAGE182,STAGE183,STAGE184,
  STAGE185,STAGE186,STAGE187,STAGE188,STAGE189,STAGE190,STAGE191,STAGE192,
  STAGE193,STAGE194,STAGE195,STAGE196,STAGE197,STAGE198,STAGE199,STAGE200,
  STAGE201,STAGE202,STAGE203,STAGE204,STAGE205,STAGE206,STAGE207,STAGE208,
  STAGE209,STAGE210,STAGE211,STAGE212,STAGE213,STAGE214,STAGE215,STAGE216,
  STAGE217,STAGE218,STAGE219,STAGE220,STAGE221,STAGE222,STAGE223,STAGE224,
  STAGE225,STAGE226,STAGE227,STAGE228,STAGE229,STAGE230,STAGE231,STAGE232,
  STAGE233,STAGE234,STAGE235,STAGE236,STAGE237,STAGE238,STAGE239,STAGE240,
  STAGE241,STAGE242,STAGE243,STAGE244,STAGE245,STAGE246,STAGE247,STAGE248,
  STAGE249,STAGE250,STAGE251,STAGE252,STAGE253,STAGE254,STAGE255,STAGE256,
  STAGE257,STAGE258,STAGE259,STAGE260,STAGE261,STAGE262,STAGE263,STAGE264,
  STAGE265,STAGE266,STAGE267,STAGE268,STAGE269,STAGE270,STAGE271,STAGE272,
  STAGE273,STAGE274,STAGE275,STAGE276,STAGE277,STAGE278,STAGE279,STAGE280,
  STAGE281,STAGE282,STAGE283,STAGE284,STAGE285,STAGE286,STAGE287,STAGE288,
  STAGE289,STAGE290,STAGE291,STAGE292,STAGE293,STAGE294,STAGE295,STAGE296,
  STAGE297,STAGE298,STAGE299,STAGE300,STAGE301,STAGE302,STAGE303,STAGE304,
  STAGE305,STAGE306,STAGE307,STAGE308,STAGE309,STAGE310,STAGE311,STAGE312,
  STAGE313,STAGE314,STAGE315,STAGE316,STAGE317,STAGE318,STAGE319,STAGE320,
  STAGE321,STAGE322,STAGE323,STAGE324,STAGE325,STAGE326,STAGE327,STAGE328,
  STAGE329,STAGE330,STAGE331,STAGE332,STAGE333,STAGE334,STAGE335,STAGE336,
  STAGE337,STAGE338,STAGE339,STAGE340,STAGE341,STAGE342,STAGE343,STAGE344,
  STAGE345,STAGE346,STAGE347,STAGE348,STAGE349,STAGE350,STAGE351,STAGE352,
  STAGE353,STAGE354,STAGE355,STAGE356,STAGE357,STAGE358,STAGE359,STAGE360,
  STAGE361,STAGE362,STAGE363,STAGE364,STAGE365,STAGE366,STAGE367,STAGE368,
  STAGE369,STAGE370,STAGE371,STAGE372,STAGE373,STAGE374,STAGE375,STAGE376,
  STAGE377,STAGE378,STAGE379,STAGE380,STAGE381,STAGE382,STAGE383,STAGE384,
**FIRST CARD OF SUBROUTINE **

**HYSORT (DAHYSO)**

---

**SUBROUTINE HYSORT**

This subroutine recomputes bed-material size distribution due to degradation/agggradation in each time period.

---

**COMMON / DMS/N, NT, M1, MAXH, YH, XYH, XGM, YGM, IM, NIP1, TONX, NOEX, NT3651, NT3652, NTRIL, NMAX, NBED, AAXB, NBED, NTP, NT1, IDREG, NTY COMMON / SCAL/IX, IY, IZ, IX1, IY1, Z1, GAMA, IFLAG, 11, INDSS, IFLAG1, 11 LIMIT, IM1, IX2, IS2, IS1, ALFA, BETA, C21, C22, PMX, IFR, IPRINT, 21UP, STR, CAR, CPPEUP, IBUG, ICND, ICFF COMMON / WATER/CM, FPCW, CPM, T1, IRF, CPEV, FF COMMON / SLOHY/S, 1, K COMMON / CPUT/TIME1 (13), CALDF (13) TIME=TSMTS (INT1, INT2) TIME1 (13) = (INT1-CPU) / 100. + TIME1 (11) CALDF (11) = CALDF (11) + 1.

**RETURN**

*******

**ENTRY**

**HYSORT**

*******

**WARNING:** ARMORING FACTOR APPROACHING UNITY

**TIME=TSMTS (INT1, INT2)**

**CPU=INT1**
IF (ACF(I) .GT. 0.98) WRITE (6, 2000) ACF(I), I, ITIME, IT
2000 FORMAT (T5,'WARNING: ARMORING FACTOR = ',F5.3, ' IN REACH',
1 I3, ' AT TIME', I6, ' IT=', I5)
ACF(I) = ADF(I) * 0.98
NA=N-1
IADJ=0
VOLOUT(I)=0
TH(I)=0
TOTAL=0
LLIM(I)=0

DA=(RA(I)+RA(I+1)) *.0.5
SM=(SP(I)+SP(I+1)) *.0.5
THA=DA*SM/(1.65*D30(I))
THA=THA*.33333

MIXED LAYER THICKNESS COMPUTED BY EQ. 6, LHR 250

TM=DA*(0.079865+2.23897*THA-18.1284*THA**2+70.90*THA**3
# .98.3293*THA**2+THA**2)*0.50*PHEL
DAE=DA*.1
IF (TM .LT. DAE) TM=DAE
PR1=TM/DA

12 IF (IBUG .EQ. 1) WRITE (6, 12) I, TH, DA, PR1
12 FORMAT (5X,'I=', I4, 'TH=', E14.6, 'DA=', E14.6, 'PR=', E14.6)

ADJUSTMENT OF SIZE DIST. IF MIXED-LAYER THICKNESS IS
MORE THAN THE TOP-LAYER SEDIMENT BED THICKNESS
INCONSISTENT WITH GLOBAL ITERATIONS

IF (NSD .EQ. 0. OR. IT .GT. 1) GO TO 315
IF (NBEL(I) .EQ. 0) GO TO 315
IF (TM .LE. THBED(I)) GO TO 315
LL=NBEL(I)
DO 300 L=1, LL
T1=THBED(I)*L
IF (TM .GT. T1) GO TO 305
300 CONTINUE
305 L2=L-1
DO 310 K=1, L2
PT(I,K)=PT(I,K) + THBED(I) / TM*PBED(I,K,L)
310 CONTINUE

IF (L2 .EQ. 0) GO TO 310
DO 306 L1=1, L2
PT(I,K)=PT(I,K) + THBED(I) / TM*PBED(I,K,L1)
306 CONTINUE
315 CONTINUE

DO 75 K=1, N1
DELT(I,K)=0
IF (IT .EQ. 1) PIU(I,K)=W2K(I,K)
75 TOTAL=TOTAL+TDELL(I,K)
DO 100 K=1, N1
T1(I,K)=TM*(1.0-P(K))*PT(I,K)
TB(I,K)=PI(I,K)*(1.-DETA*ACF(I))
100 CONTINUE
IF (IBUG .EQ. 1) WRITE (6, 56) I, K, TB(I,K), TDELD(I,K)

116
FORMAT (/, 10X, 'TB', (',', 12, ',', '11, ', '12)', 'TB', 'EB', 14.6, 2X, 'TDELD=', 'EB', 14.6)
TB(I,K) = AMAX1 (0., TB(I,K))
TH(I) = TH(I) + TB(I,K)
IF (TDELD(I,K).LT.0.0) GO TO 76
IF (IADJ.EQ.0) GO TO 72
IF (IT.EQ.1 .AND. TDELD(I,K) .GT. TB(I,K)) SSC(I,K) =
* TB(I,K)/TDELD(I,K) *** SSC(I,K) =
CONTINUE
IF (IT.EQ.1 .AND. TDELD(I,K) .GT. TB(I,K)) LLIM(I) = 1
IF (IT.EQ.1 .AND. TDELD(I,K).NE.0) DELT(I,K) = IDELT*
* TB(I,K)/TDELD(I,K)
IF (TDELD(I,K).EQ.0) DELT(I,K) = 900.0
DELT(I,K) = AMIN1 (DELT(I,K), 9000.0)
IF (IT.EQ.1 .AND. TDELD(I,K) .GT. TB(I,K)) TDELD(I,K) = TB(I,K)
IF (IT.EQ.1) GO TO 70
ABC = TH(I) * FTP(I,K)
IF (IADJ.EQ.0) GO TO 74
IF (TDELD(I,K).GT.ABC) SSC(I,K) = ABC/TDELD(I,K) *** SSC(I,K) =
CONTINUE
IF (TDELD(I,K).GT.ABC) LLIM(I) = 1
IF (TDELD(I,K).NE.0) DELT(I,K) = IDELT*ABC/TDELD(I,K)
DELT(I,K) = AMIN1 (DELT(I,K), 9000.0)
TDELD(I,K) = AMIN1 (TDELD(I,K), ABC)
IF (TDELD(I,K).GE.0.0) GO TO 90
CONTINUE
DELT(I,K) = AMIN1 (DELT(I,K), 9000.0)
IF (IT.EQ.1) GO TO 90
VOLOUT(I) = VOLOUT(I) + TDELD(I,K)
CONTINUE
IF (IADJ.EQ.0) GO TO 125
CTO(I) = 0.
DO 120 K = 1, N1
CTO(I) = CTO(I) + SSC(I,K)
IF (IBUG.EQ.1) WRITE (6, 16) VOLOUT(I), I, J
16 FORMAT (5X, '*** VOLOUT = ', EB, 14.7, 2X, 'I=', '1, ', '12, ', '2X, 'J=', '1, ', '12)
CONTINUE
IF (VOLOUT(I).LT.0.0) GO TO 300
CCC = 0
IF (IT.EQ.1) TH1(I) = TH(I)
DIN = TH(I) + VOLOUT(I) - TH1(I)
EL11 = TALWG(I) - TH1(I)
EL21 = EL11 - DIN
BEL = BMI(I)
IF (IT.EQ.1) BEL = EL21
DO 200 K = 1, N1

C
LINDEX = 0
LDEP = 0
IF (IBED.EQ.1) CALL VSORT (BEL, EL11, EL21, DIN, PTT, PT, PTU, VOLIN,
# NBEL, THBED, PBED, LINDEX, TALWG, CDEP, LDEP)
C
IF (LINDEX.EQ.1) GO TO 135
IF (IT.EQ.1) GO TO 130
IF (EL21.GT.EL11) GO TO 130
IF (BEL .GE. EL11) VOLIN(I,K) = DIN*PTT(I,K)
IF (EL11.GE.BEL .AND. BEL .GT. EL21) VOLIN(I,K) = (EL11-BEL)
# *PTU(I,K) + BEL .LT. EL21) *PTI(I,K)
IF (EL21.GT.BEL) VOLIN(I,K) = DIN*PTU(I,K)

117
IF (IT.EQ.1) VOLIN(I,K)=DIN*PT1(I,K)
IF (EL2I.GT.ELII) VOLIN(I,K)=DIN*PTF(I,K)

CONTINUE
VOLIN(I,K)=AMAX1(VOLIN(I,K),0.0)
BZ(I,K)=TH(I)*PTF(I,K)-TDELD(I,K)*VOLIN(I,K)
BZ(I,K)=AMAX1(BZ(I,K),0.0)
CCC=CCC+BZ(I,K)
EL1(I)=EL1I
EL2(I)=EL2I
EMIN=EL1I
IF (EL2I.LT.ELII) ELIN=EL2I
BEL=AMIN1(BEL,EMIN)
TH(I)=0
DO 220 K=1,N1
219 TH(I)=TH(I)+TB(I,K)
PT(I,K)=BZ(I,K)/CCC
CONTINUE
BML(I)=BEL
GO TO 900

CC1=0
IF (IT.EQ.1) TH(I)=TH(I)
DIN=TH(I)-VOLOUT(I)-TH(I)
ELII=TALWV(I)-TH(I)
EL2I=ELII+DIN
BEL=BML(I)
IF (IT.EQ.1) BEL=ELII
DO 950 K=1,N1

C
LININDEX=0
LDEP=1
IF (IBLD.EQ.1) CALL VSORT(BEL,EL1I,EL2I,DIN,PTT,PTF,PTU,VOLIN,
# NEEL,TB,PTU,PTF,EL1I,CC1,ELII,TALWV,COLP,CELP)
C
IF (LININDEX.EQ.1) VOLIN(I,K)=-VOLIN(I,K)
IF (LININDEX.EQ.1) GO TO 632
IF (IT.EQ.1) GO TO 530
IF (EL2I.GT.ELII) GO TO 330
IF (BEL.LT.EL2I) VOLIN(I,K)=DIN*PTT(I,K)
IF (ELII.GE.BEL.AND.BEL.GT.EL2I) VOLIN(I,K)=-(ELII-BEL)
# *PTT(I,K)+(BEL-EL2I)*PTF(I,K))
IF (EL2I.GT.BEL) VOLIN(I,K)=DIN*PTF(I,K)
830 IF (IT.EQ.1) VOLIN(I,K)=DIN*PTF(I,K)
IF (EL2I.GT.ELII) VOLIN(I,K)=DIN*PTP(I,K)
832 CONTINUE
VOLIN(I,K)=AMAX1(VOLIN(I,K),0.0)
BZ(I,K)=TH(I)*PTF(I,K)-TDELD(I,K)-VOLIN(I,K)
BZ(I,K)=AMAX1(BZ(I,K),0.0)
CC1=CC1+BZ(I,K)
EL1(I)=EL1I
EL2(I)=EL2I
EMIN=EL1I
IF (EL2I.LT.ELII) ELIN=EL2I
IF (EMIN.LT.EL) BEL=EMIN
TH(I)=0.
DO 860 K=1,N1
858 TH(I)=TH(I)+TB(I,K)
PT(I,K)=BZ(I,K)/CC1
CONTINUE
\[ B_M(1) = B_E \]
\[ \text{TIME} = \text{TSTEPS} \times (\text{INT1}, \text{INT2}) \]
\[ \text{TIME1} = \text{INT1} \times \text{CP0} / 100 + \text{TIME1} \]
\[ \text{CALDF} = \text{CALDF} + 1 \]

RETURN

\* FIRST CARD OF SUBROUTINE TRIB *

SUBROUTINE TRIB (QTR, QSTR, TDELTA, LOCTR, PTTRB, AC, BC, Q, 1)
  LOCBER, DEMOS, PLAN, CTO, INPUT)

DIMENSION QTR (NTTRB), QSTR (NTTRB), TDELTA (NTTRB, N1),
  1 LOCTR (NTTRB), PTTRB (NTTRB, N1), AC (NTTRB), BC (NTTRB, Q (N),
  2 LOCBER (NBBANK), DEMOS (NBBANK), PLAN (NBBANK, N1), CTO (N)
COMMON/DMIN,N1,NT,HI,KMAX,NN,MEMO,NX,IGK,NIP1,NTONX,NOBS,
  1 NT3651, NT3052, NTTRB, NBBANK, ISED, MAXBED, NBED, NTP, NT, ILREJ, NTY
COMMON/SCALE/INDEX,L6, TDELTA, VLS, ITIME, GAMA, IFLAG, IT, INLSS, IFLAG1,
  1 ILLIMIT, MA1, IEQ, IRES, ISED, ALFA, BETA, C21, C22, FMIN, IFRA, IPRT,
  2 UP, STR, TANK, CPMUP, IBDG, ICHL, LCOFF
COMMON/BANK/IBSED, QMIN, IUPS
COMMON/CPU/INT1(18), CALDF(18)

IBSED = INDEX VARIABLE FOR INDICATING SIZE DISTR. OF BANK EROSION;
  IBSED = 0 FOR ASSUMING SAME DISTR. AS BED MATERIAL; = 1
  FOR SPECIFYING THEIR VALUES AS INPUT
QMIN = MINIMUM WATER DISCHARGE (CFS) ABOVE WHICH BANK EROSION
  OCCURS
IUPS = INDEX VARIABLE TO SPECIFY UPSTREAM SEDIMENT INFLOW;
  IUPS = 0 FOR CONSTANT SED. CONCN.; = 1 FOR FUNCTION
  OF WATER DISCHARGE
NOTE: UPSTREAM SEDIMENT INFLOW IS TREATED IN THE SAME WAY
  AS FOR TRIBUTARIES

READ (5, 10) (LOCTR (12), 12 = 1, NTTRB)
LOCTR (1) = n
WRITE (6, 15)
WRITE (6, 20) (LOCTR (12), 12 = 1, NTTRB)
FORMAT (1215)
FORMAT ('/', 10X, 'TRIBUTARY LOCATIONS (NODE NUMBERS) :', '//')
FORMAT ('/', 5X, 'MEAN SEDIMENT CONCN. AT U/S BOUNDARY=', 'F7.1,
  1 2X, 'P.P.-M.', '//')
FORMAT (15X, 10F18)
FORMAT ('/', 10X, 'TRIBUTARY WATER DISCHARGES (CFS) :', '//')
FORMAT (6P10.2)
FORMAT (15X, 10P9.0)

READ TRIB, SED, DISCH. COEFFICIENTS AC, BC IN QS = AC * Q ** BC
QS IN TONS/DAY, Q IN CFS.

DO 102 I2 = 1, NTTRB
IF (IC12 .GT. 1) GO TO 101
IF (INSS .EQ. 0) GO TO 102
IF (IUPS .EQ. 0) GO TO 102
101 READ (5, 35) (AC(I), EC(I))
102 CONTINUE
WRITE (6, 36)
DO 104 K = 1, 12
IF (IC12 .GT. 1) GO TO 103
IF (INSS .EQ. 0) GO TO 104
IF (IUPS .EQ. 0) GO TO 104
103 WRITE (6, 40) (AC(I), EC(I))
104 CONTINUE
35 FORMAT (2F14.9)
36 FORMAT (/,, 10X, 'SEDIMENT DISCHARGE COEFFICIENTS (FOR TRIBUTARIES)',
& 'S') :/,,/
40 FORMAT (15X, 2F14.3)
C
C
C
READ SIZE DISTRIBUTION OF TRIBUTARY SEDIMENT INFLOWS
C
410 WRITE (6, 28)
DO 500 K = 1, 12
READ (5, 25) (P(K), K = 1, 12)
500 CONTINUE
WRITE (6, 32) (P(K), K = 1, 12)
28 FORMAT (/,, 10X, 'SIZE DISTRIBUTION OF TRIBUTARY SEDIMENT INFLOWS :',/,,)
32 FORMAT (15X, 10F9.3)
C
C
READ PARAMETERS FOR BANK EROSION
C
IP(NBANK .EQ. 0) GO TO 900
READ (5, 61) IBSED, QMIN
WRITE (6, 65) IBSED, QMIN
READ (5, 19) (LOCER(I), I = 1, NBANK)
WRITE (6, 70)
WRITE (6, 20) (LOCER(I), I = 1, NBANK)
IP(IBSED .EQ. 0) GO TO 700
WRITE (6, 69)
DO 600 I = 1, NBANK
READ (5, 25) (PBK(I), I = 1, NBANK)
600 CONTINUE
61 FORMAT (15X, F10.1)
65 FORMAT (/,, 10X, 'IBSED=', F12, '4X, 'QMIN=', F10.1, ' CFS', ' ,/,,)
70 FORMAT (/,, 10X, 'LOCATION OF BANK EROSION (REACH NUMBER) :',/,,)
C
C
READ BANK EROSION RATE, BFRS (IN CFT./MILE/DAY)
C
READ (5, 25) BFRS(I), I = 1, NBANK
WRITE (6, 72)
WRITE (6, 74) BFRS(I), I = 1, NBANK
72 FORMAT (/,, 10X, 'BANK EROSION COEFFICIENT (CFT/MILE/DAY) :',/,,)
74 FORMAT (15X, 8F10.1)
900 CONTINUE

TIME = TIMES (INT1, INT2)
TIME1(2) = (INT1 - CF0) / 100. + TIME1(2)
CALDF (2) = CALDF (2) + 1.

RETURN
C
ENTRY TRIBOS (QSUPS)
C
TIME=ISTEPS(INIT1,INT2)
CPU=INT1
C
TRIBOS CALLED AT BEGINNING OF EACH TIME STEP TO LOAD
QSTR, TRIBUTARY SEDIMENT INFLOWS
C
COMPUTING SED. DISCH. OF TRIBUTARIES
C
DO 400 IZ=1,NTRIB
IF (IZ.GT.1) GO TO 240
I=LOCTR (IZ)
IF (INDSS.EQ.1.AND.IUPS.EQ.0) CTO(I)=QSUPS/(2.65E6)
240 CONTINUE
IF (IZ.GT.1) GO TO 250
IF (INDSS.EQ.0) GO TO 300
IF (INDSS.EQ.1.AND.IUPS.EQ.0) QSTR(IZ)=QSUPS/2.65E6
3 QTR(IZ)
IF (INDSS.EQ.1.AND.IUPS.EQ.0) GO TO 300
250 QSTR(IZ)=AC(IZ)*QTR(IZ)**(BC(IZ))*2000.0/(62.5*86400.0)
IF (IZ.GT.1) GO TO 300
I=LOCTR (IZ)
IF (INDSS.EQ.1.AND.IUPS.EQ.1) CTO(I)=QSTR(IZ)/QTR(IZ)
300 CONTINUE
400 CONTINUE
IF (INBUG.NE.0) WRITE (6,2000) (QSTR(IZ),IZ=1,NTRIB)
2000 FORMAT (9X,'TRIBUTARY SEDIMENT LOADS:',
 TIME=ISTEPS(INIT1,INT2)
 TIME1(2)=TIME1(2)+(INT1-CPU)/100.
 CALDF(2)=CALDF(2)+1.
 RETURN
END
C
********** FIRST CARD OF SUBROUTINE SEDBED **********
C
SUBROUTINE SEDBED (NBEL,THBED,PEED)
C
TIME=ISTEPS(INIT1,INT2)
CPU=INT1
C
THIS SUBROUTINE READS ADDITIONAL SEDIMENT CHARACTERISTICS
IN CASE OF VERTICAL VARIATION OF ORIGINAL BED MATERIAL
C
NBEL(I)= NO. OF ELEVATIONS AT WHICH SED. SIZE DISTR. CHANGES
AT REACH I
THBED(I)= THICKNESS OF HOMOGENEOUS SED. SIZE DISTR. IN REACH I
PEED(I,K,L)= SED. SIZE DISTR. (IN FRACTION) AT REACH I,
 FRACTION K IN SEDIMENT LAYER L
C
DIMENSION NBEL(NN),THBED(NN),PEED(NN,N1,NAXEBED)
COMMON/DIAS/N,N1,N1,T,H1,NAVBA,NN,REMO,NAI,INT1,NIOA1,NOBS,
 1 NT3651,NT3652,NTRIB,NDAMB,IBED,INBED,NTF,NT1,INBED,NTY
 COMMON/CPU/TIME1(18),CALDF(16)
C
READ (5,10) (NBEL(I),I=1,NN)
WRITE (6,15)
C
READ (5,20) (THBED(I),I=1,NN)
WRITE (6,21)
C
WHITE (6, 20) (NBEL(I), I=1, N)

C
READ (5, 25) (TBED(I), I=1, N)
WRITE (6, 30)
WRITE (6, 35) (TBED(I), I=1, N)

C
WRITE (6, 40)
DO 100 I=1, N
LL=NBEL(I)
IF (LL.EQ.0) GO TO 100
DO 100 L=1, LL
READ (5, 25) (FBED(I, K, L), K=1, N1)
WRITE (6, 35) (FBED(I, K, L), K=1, N1)
CONTINUE

C
10 FORMAT (12I5)
15 FORMAT (/6X, 'NBEL(I) : ',/)
20 FORMAT (10X, 12I5)
25 FORMAT (6F10.3)
30 FORMAT (/6X, 'TBED(I) : ',/)
35 FORMAT (10X, 6F10.3)
40 FORMAT (/6X, 'FBED(I, K, L) : ',/)

C
TIME=STEPS(INIT1, INIT2)
TIME1(3)=TIME1(3) + (INIT1-CPU)/100.
CALDF(3)=CALDF(3)+1.
RETURN
END

C

SUBROUTINE DACHAN (ICHET, ICOFFT, ICHBL, ICOFFL, REACH,51,
& AREA, NCHBL, NCOFFL, MA, R1)

C
TIME=STEPS(INIT1, INIT2)
CPU=INIT1

C

THIS SUBROUTINE READS AND COMPUTES THE EFFECTS OF CHANNEL-WIDTH
CHANGES AND CHANNEL CUTOFF AS FUNCTION OF TIME

C

DIMENSION ICHET(NT3651), ICOFFT(NT3651), ICHBL(NT3651, N),
& ICOFFL(NT3651, N), B1(N, MAXA), REACH(N), AREA(N, MAXA),
& NCHBL(NT3651), NCOFFL(NT3651), MA(N), R1(N, MAXA)
COMMON/IMS/HN, N1, NT, N1, MAXA, NN, B1, N, MAXA, N, REACH(N),
& NT3651, NT3652, WRITE, NEAK, ISD, MAXBED, NEED, NTP, NT1, IDRED, NTY
& COMMON/SCALE/INDEX, LB, IDLE, VIS, TIME, GAMM, IFLAG, IT, INESS, IFLAG1,
& 1 LIMIT, H1, I1, IES, ISD, ALFA, BETA, C21, C22, FRIX, IFR, IPRINT,
& 2IF, STR, CARS, CP12D, LBD, ICEB, ICOFF
COMMON/CPU/TIME1(10), CALDF(10)

C
ICHBT(IT)=TIME INTERVALS IN WHICH CHANNEL WIDTHS ARE CHANGED
ICOFFT(IT)=TIME STEPS IN WHICH CUTOFFS ARE INCORPORATED
ICHBL(NI)=LOCATION (NODE NO.) OF CHANNEL-WIDTH CHANGES
IN TIME INTERVAL IT
ICOFFL(NI)=LOCATION (REACH NO.) OF CUTOFFS IN TIME STEP IT
NCOFFT=TOTAL NO. OF TIME STEPS IN WHICH CUTOFFS ARE MADE
NCOFFL(I)= TOTAL NO. OF CUTOFF-REACHES IN TIME STEP IT
NCHBT= TOTAL NO. OF TIME STEPS IN WHICH CHANNEL WIDTHS
NCHBL(I1) = TOTAL NO. OF NODES OF CHANNEL-WIDTH CHANGES IN TIME STEP I1

READ PARAMETERS FOR CHANGING WIDTH

IF (ICHBT.EQ.0) GO TO 100
WRITE (6, 19)
READ (5, 10) NCHBT
READ (5, 10) (NCHBL(I1), I1=1,NCHBT)
WRITE (6, 12) NCHBT
WRITE (6, 14)
WRITE (6, 15) (NCHBL(I1), I1=1,NCHBT)
FORMAT (1215)

FORMAT (/, 10X, 'NCHBL=', I3, ')

FORMAT (/, 5X, 'ICHBT(I1)', ')

FORMAT (15X, 1215)

READ (5, 10) (ICHBT(I1), I1=1,NCHBT)
WRITE (6, 1b)
WRITE (6, 15) (ICHBT(I1), I1=1,NCHBT)
WRITE (6, 16)
DO 50 I1=1,NCHBT
L=NCHBL(I1)
READ (5, 10) (ICHBL(I1, I), I=1,L)
WRITE (6, 15) (ICHBT(I1, I), I=1,L)
FORMAT (/, 5X, 'ICHBT(I1)', ')

FORMAT (/, 5X, 'ICHBL(I1, I)', ')

FORMAT (/, 10X, 'INPUT VALUES FOR CHANNEL-WIDTH CHANGES WITH ',

' TIME : ', ')

CONTINUE

READ PARAMETERS FOR CHANNEL CUTOFF

IF (ICOFF.EQ.0) GO TO 200
WRITE (6, 29)
READ (5, 10) NCOFFT
READ (5, 10) (NCOFFFL(I1), I1=1,NCOFFT)
WRITE (6, 22) NCOFFT
WRITE (6, 24)
WRITE (6, 15) (NCOFFFL(I1), I1=1,NCOFFT)
FORMAT (/, 10X, 'NCOFFT=', I3, ')

FORMAT (/, 5X, 'NCOFFFL(I1)', ')

READ (5, 10) (ICOFFT(I1), I1=1,NCOFFT)
WRITE (6, 26)
WRITE (6, 15) (ICOFFT(I1), I1=1,NCOFFT)
WRITE (6, 28)
DO 150 I1=1,NCOFFT
L=NCOFFL(I1)
READ (5, 10) (ICOFFL(I1, I), I=1,L)
WRITE (6, 15) (ICOFFL(I1, I), I=1,L)
FORMAT (/, 5X, 'ICOFFT(I1)', ')

FORMAT (/, 5X, 'ICOFFL(I1, I)', ')

FORMAT (/, 10X, 'INPUT VALUES FOR CHANNEL CUTOFF : ', ')

CONTINUE

ADJUSTMENT FOR CHANNEL-WIDTH CHANGES
C

TIME=TSTMP$ (INT1, INT2)
TIME1 (15) = (INT1-CPU)/100 + TIME1 (15)
CALDF (15) = CALDF (15) + 1.

RETURN

C

***************
ENTRY CHANGE
C

***************

TIME=TSTMP$ (INT1, INT2)
CPU=INT1
IF (ICH = EQ. 0) GO TO 500
ITEST=0
DO 210 IT=1, NCHET
IT= ICHET (IT)
IF (IT .EQ. IT) ITEST= ITEST + 1
IF (IT .EQ. IT) GO TO 215

210 CONTINUE

215 IF (ITEST .EQ. 0) GO TO 500
L= NCHET (IT)
DO 220 IT=1, L
I= ICHET (IT, I)
M= M (I)
READ (5, 30) (B1 (I, MAI), MAI=1, NN)
WRITE (6, 32) IT, I
WRITE (6, 35) (B1 (I, MAI), MAI=1, NN)
DO 220 MAI=1, NN
AREA (I, MAI) = B1 (I, MAI) * K1 (I, B1)

220 CONTINUE

30 FORMAT (6F10.1)

32 FORMAT ('// 10X,60 ("*")', //, 15X, 'NEW CHANNEL WIDTHS (FT.)',
',
', 'IT=', I, '13, 2X, 'I=', I, '13, //, 10X, 60 ("*")', //)

35 FORMAT (10X, GF10.1)

500 CONTINUE

C

ADJUSTMENT FOR CHANNEL CUTOFF
C

IF (ICOFF .EQ. 0) GO TO 1000
ITEST=0
DO 310 IT=1, ICOFF
IT= ICOFF (IT)
IF (IT .EQ. IT) ITEST= ITEST + 1
IF (IT .EQ. IT) GO TO 315

310 CONTINUE

315 IF (ITEST .EQ. 0) GO TO 1000
L= ICOFF (IT)
DO 320 IT=1, L
I= ICOFF (IT, I)
READ (5, 30) REACH (I)
WRITE (6, 42) IT, I
WRITE (6, 35) REACH (I)

320 CONTINUE

C

42 FORMAT ('//, 10X, 65 ("*")', //, 15X, 'NEW REACH LENGTH (FT.) FOR',
',
', 'CUTOFF AT IT=', I, '13, 2X, 'REACH=', I, '13, //, 10X, 65 ("*")', //)

1000 CONTINUE

TIME=TSTMP$ (INT1, INT2)
TIME1 (15) = TIME1 (15) + (INT1-CPU)/100.
CALDF (15) = CALDF (15) + 1.

124
RETURN
END

SUBROUTINE VSORT (EL1, EL21, DNT, PTT, PT, PTU, VOLIN,
                   NBEL, THBED, PEED, LINDEX, STAGE1, CDEP, LDEP)

TIME=TSTEPS(INT1, INT2)
CPU=INT1

THIS SUBROUTINE ADJUSTS MIXED-LAYER COMPOSITION IN CASE OF
VERTICAL VARIATION IN SEDIMENT-LID SIZE DISTRIBUTION

DIMENSION PTT (N1, N1), PT (N1, N1), PTU (N, N1), VOLIN (N, N1),
1 NBEL (N1), THBED (N1), PEED (N1, N1, MAAXED), STAGE1 (N1, MAAXA),
2 CDEP (N)

COMMON/LMS/N, N1, M1, MAAXA, MAAXN, NEXO, NX, IGR, N1P1, NTONX, NOBS,
1 NT3651, NT3652, NTRIB, NQANK, LBL, MAAXED, NBED, NTE, NT1, IDHEJ, NTY
COMMON/SCAL/NINDEX, LO, IDELT, VIS, TIME, GAMA, IFLAG, IT, INDSS, IFLAG1,
1 LIM1, LI1, LE1, TS1, ALFA, BETA, C21, C22, PMIX, IFR, IPRINT,
2 IMF, STR, CARB, CPM, SDF, IBUG, IOBE, ICOFF

COMMON/SLO MYS/I1, K

COMMON/CP1/1, TIME1(10), C ALDF(10)

IF (NBEL(I) .EQ. 0) GO TO 1000
IF (LDEP .EQ. 1) DNM = -DIN
INDEX = 1
NB = NBEL(I)
L1 = 0
L2 = 0
LB = 0

IF (I.LEQ.1) CDEP = CDEP(I)
IF (I.GT.1) CDEP = (CDEP(I) + CDEP(I-1))/2.0
DO 200 L = 1, NB
   T1 = STAGE1(I, I) - L*THBED(I) + CDEP(I)
   IF (NB .GT. 1) GO TO 50
   IF (T1.GT.EL2I) L1 = L
   IF (T1.GT.EL21) L2 = L
   IF (T1.GT.BEL) LB = L
   IF (K .EQ. 1) AND (IBUG .EQ. 1) WRITE (6, 49) I, EL1, EL21, SEL, T1
49 FORMAT (5X, I1 = 'EL1=', X14.7, 2X, 'EL21=', X14.7, 2X, 'SEL=', X14.7)
200 CONTINUE

IF (NB .EQ. 1) GO TO 200
T2 = STAGE1(I, I) - (L1 + 1)*THBED(I) + CDEP(I)
IF (EL1.LT.T1.AND.EL1.LT.T2) L1 = L
IF (EL1.LT.T1.AND.L.EQ.NB) L1 = L
IF (EL21.LT.T1.AND.EL21.LT.T2) L2 = L
IF (EL21.LT.T1.AND.L.EQ.NB) L2 = L
IF (BEL.LT.T1.AND.BEL.GT.T2) LC = L
IF (BEL.LT.T1.AND.L.EQ.NB) LC = L

200 CONTINUE

IF (L1.GT.0.AND.L2.GE.L1) PTU(I, K) = PEED(I, X, L1)

LL = L2 - 1
IF (EL21.GT.BEL) GO TO 990
IF (EL21.GT.EL1) GO TO 990
IF (BEL.LT.EL11) GO TO 250
TIME=TSTMPS (INT1,INT2)
TIME1(14)=TIME1(14)+(INT1-CPU)/100.
CALDF(14)=CALDF(14)+1.
RETURN
END

FIRST CARD OF SUBROUTINE SHIELD

SUBROUTINE LASHIL

TIME=TSTMPS (INT1,INT2)
CPU=INT1

DIMENSION SHP1(50),SHP2(500)
COMMON/CPU/TIME1(16),CALDF(16)
COMMON/SED/RS,SHP

INITIAL LOADING OF SHIELD TABLES

DO 10 I=1,50
   RS=I*0.1
   IF (RS.GE.1.AND.RS.LE.2.0) SHPP=118*(RS**(-.973))
   IF (RS.GT.2.0.AND.RS.LE.4.0) SHPP=900*(RS**(-.585))
   IF (RS.GT.4.0.AND.RS.LE.10.0) SHPP=434*(RS**(-.191))
   SHP1(I)=SHPP
10 CONTINUE

DO 20 I=5,500
   RS=I
   IF (RS.GT.4.0.AND.RS.LE.10.0) SHPP=434*(RS**(-.191))
   IF (RS.GT.10.0.AND.RS.LE.50.0) SHPP=275*(RS**(.0792))
   IF (RS.GT.50.0.AND.RS.LE.500.0) SHPP=194*(RS**(.181))
   SHP2(I)=SHPP
20 CONTINUE

TIME=TSTMPS (INT1,INT2)
TIME1(8)=TIME1(8)+(INT1-CPU)/100.
CALDF(8)=CALDF(8)+1.
RETURN

ENTRY SHIELD

TIME=TSTMPS (INT1,INT2)
CPU=INT1
IF (RS.GT.5.) 40,50,50
40 INDEX1=10*AMAX1(0.1,RS)
SHP=SHP1(INDEX1)
GO TO 60
50 INDEX2=AMAX1(RS,500.)
SHP=SHP2(INDEX2)
60 TIME=TSTMPS (INT1,INT2)
TIME1(8)=TIME1(8)+(INT1-CPU)/100.
CALDF(8)=CALDF(8)+1.
RETURN
END

FIRST CARD OF SUBROUTINE ERROR1

SUBROUTINE ERROR1(TEND,ZERO,N,41,N1,NT,MAXA,NOES,NX,IGN)

TIME=TSTMPS (INT1,INT2)

CPU=INT1
COMMON/CPUT/TIME1(18),CALDF(18)

C IERR=1
WRITE(6,2000) IERR
2000 FORMAT(1X,'T10.3',1X,'ERROR',1X)
WRITE(6,2001) IEND,IN10,N,IN1,NT,NMAX,NOSB,NX,IGR
2001 FORMAT(T20,'REQUIRED MEMORY=','T17',1X,'(WORDS) EXCEEDS DIMENSION',1X,
'OF ARRAY T(',17,')',1X,T20,'N,M1,N1,NT,NMAX,NOSB,NX',1X)
2 'IGR=',I8.5)
TIME=TSTEPS(IN1,IN12)
TIME1(6)=(IN1-CPU)/100.+TIME1(6)
CALDF(6)=CALDF(6)+1.
STOP

C ENTRY ERROR2(IN,ME,NMAX)
TIME=TSTEPS(IN1,IN12)
CPU=INT1
IERR=2
WRITE(6,2000) IERR
WRITE(6,2002) IN,ME,NMAX
2002 FORMAT(T20,'SECTION',13,1X,'NO. OF DEFINITION LEVELS=',13,
'EXCEEDS MAXIMUM ALLOCATED',1X,NMAX=',',I4)
TIME=TSTEPS(IN1,IN12)
TIME1(6)=(IN1-CPU)/100.+TIME1(6)
CALDF(6)=CALDF(6)+1.
STOP

C ***************
ENTRY ERROR3(IT,ITIME,TEMPF,THLAST)
C ***************
TIME=TSTEPS(IN1,IN12)
CPU=INT1
IERR=3
WRITE(6,2000) IERR
WRITE(6,2003) TEMPF,THLAST,IT,ITIME
2003 FORMAT(T20,'WATER TEMPERATURE=',F10.3,1X,'EXCEEDS MAXIMUM ',1X,
'ALLOWED=',F10.3,1X,IT=',I5,1X,ITIME=',I5)
TIME=TSTEPS(IN1,IN12)
TIME1(6)=(IN1-CPU)/100.+TIME1(6)
CALDF(6)=CALDF(6)+1.
RETURN

C ***************
ENTRY ERROR4(1,IT,ITIME,ITER,DEL)
C ***************
TIME=TSTEPS(IN1,IN12)
CPU=INT1
IERR=4
WRITE(6,2000) IERR
WRITE(6,2004) ITER,1,IT,ITIME,DEL
2004 FORMAT(T20,'BACKWARD ITERATIONS EXCEED',14,1X,'SECTION',14,
'IT=',I4,1X,ITIME=',I4,1X,DELTA=',F10.3)
TIME=TSTEPS(IN1,IN12)
TIME1(6)=(IN1-CPU)/100.+TIME1(6)
CALDF(6)=CALDF(6)+1.
RETURN

C ***************
ENTRY ERROR5(1,IT,ITIME,IT2,TT)
C ***************

128
TIME = TSTEPS (INT1, INT2)
CPU = INT1

IERR = 5
WRITE (6, 2000) I.ERR
WRITE (6, 2005) IT2, IT, ITIME, IT

2005 FORMAT (I20, 'TLM ITERATIONS EXCEED', I4, ' SECTION', I4, 'IT=',
I4, ' ITIME=', I4, ' TT=', F10.3)
TIME = TSTEPS (INT1, INT2)
TIME1 (6) = TIME1 (6) + (INT1 - CPU) / 100.
CALDF (6) = CALDF (5) + 1.

RETURN
END
APPENDIX D.

Sample Data Set
SAMPLE DATA FOR MISSOURI RIVER MODEL

<table>
<thead>
<tr>
<th></th>
<th>30</th>
<th>82000</th>
<th>16</th>
<th>8</th>
<th>6</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>5</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1800</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>15</td>
<td>7200</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>0.50</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.00189000</td>
<td>0.005</td>
<td>0.5</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>29</td>
<td>27</td>
<td>22</td>
<td>16</td>
<td>15</td>
<td>12</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.002652</td>
<td>1.5790</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.002652</td>
<td>1.5790</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.001360</td>
<td>1.0700</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.0002649</td>
<td>2.1220</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.0001963</td>
<td>2.8533</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.00002854</td>
<td>2.6633</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.00004542</td>
<td>2.1422</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.402</td>
<td>0.46</td>
<td>1.23</td>
<td>0.12</td>
<td>0.003</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>0.39</td>
<td>0.433</td>
<td>1.27</td>
<td>0.05</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.39</td>
<td>0.433</td>
<td>1.27</td>
<td>0.05</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.39</td>
<td>0.433</td>
<td>1.27</td>
<td>0.05</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.65</td>
<td>0.284</td>
<td>0.035</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.65</td>
<td>0.284</td>
<td>0.035</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.00</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.402</td>
<td>0.46</td>
<td>1.23</td>
<td>0.12</td>
<td>0.003</td>
<td>0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>657</td>
<td>657</td>
<td>1170</td>
<td>1170</td>
<td>1732</td>
<td>1732</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.062</td>
<td>0.149</td>
<td>0.297</td>
<td>0.590</td>
<td>1.19</td>
<td>2.4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.60</td>
<td>9.52</td>
<td>19.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

SECTION NO. 1

859.85 | 600.000 |
895.85 | 600.0 |
0.000 | 0.04 | 0.71 | 0.97 | 0.99 | 0.996 |
0.998 | 0.9985 | 1.00 |
533.0 | 2 |

SECTION NO. 2

869.85 | 600.0 |
899.85 | 600.0 |
0.000 | 0.04 | 0.71 | 0.97 | 0.99 | 0.996 |
0.998 | 0.9986 | 1.00 |
543.0 | 2 |

SECTION NO. 3

879.85 | 600.0 |
909.85 | 600.0 |
0.000 | 0.04 | 0.71 | 0.97 | 0.99 | 0.996 |
0.998 | 0.9986 | 1.00 |
553.0 | 2 |

SECTION NO. 4

889.85 | 600.0 |
919.85 | 600.0 |
0.000 | 0.04 | 0.71 | 0.97 | 0.99 | 0.996 |
0.998 | 0.9986 | 1.00 |
561.0 | 2 |

SECTION NO. 5

900.82 | 675.0 |
930.82 | 675.0 |
<p>| | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0.000</td>
<td>0.10</td>
<td>0.85</td>
<td>0.973</td>
<td>0.9915</td>
<td>0.996</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9978</td>
<td>0.9987</td>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>571.0</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>SECTION NO. 6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>910.25</td>
<td>575.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>940.25</td>
<td>575.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000</td>
<td>0.016</td>
<td>0.20</td>
<td>0.57</td>
<td>0.802</td>
<td>0.944</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.986</td>
<td>0.9964</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>581.0</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>SECTION NO. 7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>919.93</td>
<td>575.0</td>
<td>575.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>949.93</td>
<td>575.0</td>
<td>575.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000</td>
<td>0.018</td>
<td>0.20</td>
<td>0.706</td>
<td>0.92</td>
<td>0.962</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9946</td>
<td>0.9979</td>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>591.0</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>SECTION NO. 8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>930.12</td>
<td>550.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>960.12</td>
<td>550.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000</td>
<td>0.086</td>
<td>0.80</td>
<td>0.9765</td>
<td>0.9913</td>
<td>0.9962</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9934</td>
<td>0.9983</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>600.0</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>SECTION NO. 9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>938.39</td>
<td>575.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>968.39</td>
<td>575.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000</td>
<td>0.03</td>
<td>0.88</td>
<td>0.97</td>
<td>0.9892</td>
<td>0.995</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9973</td>
<td>0.9984</td>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>610.5</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>SECTION NO. 10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>946.80</td>
<td>600.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>976.80</td>
<td>600.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000</td>
<td>0.07</td>
<td>0.79</td>
<td>0.97</td>
<td>0.99</td>
<td>0.9955</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9975</td>
<td>0.9986</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>620.0</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>SECTION NO. 11</td>
<td></td>
<td></td>
</tr>
<tr>
<td>956.11</td>
<td>600.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>986.11</td>
<td>600.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000</td>
<td>0.07</td>
<td>0.76</td>
<td>0.968</td>
<td>0.9897</td>
<td>0.9955</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9976</td>
<td>0.9985</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>630.0</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>SECTION NO. 12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>965.13</td>
<td>600.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>995.13</td>
<td>600.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000</td>
<td>0.032</td>
<td>0.38</td>
<td>0.986</td>
<td>0.998</td>
<td>0.995</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9975</td>
<td>0.9985</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>640.0</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>SECTION NO. 13</td>
<td></td>
<td></td>
</tr>
<tr>
<td>973.78</td>
<td>600.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1003.78</td>
<td>600.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000</td>
<td>0.022</td>
<td>0.39</td>
<td>0.95</td>
<td>0.9367</td>
<td>0.9948</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9973</td>
<td>0.9984</td>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>648.2</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>SECTION NO. 14</td>
<td></td>
<td></td>
</tr>
<tr>
<td>981.06</td>
<td>600.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1011.06</td>
<td>600.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000</td>
<td>0.018</td>
<td>0.30</td>
<td>0.85</td>
<td>0.968</td>
<td>0.99</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9981</td>
<td>0.9982</td>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>658.5</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>SECTION NO. 15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>988.99</td>
<td>650.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1018.99</td>
<td>650.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000</td>
<td>0.04</td>
<td>0.44</td>
<td>0.94</td>
<td>0.994</td>
<td>0.994</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9971</td>
<td>0.9985</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>669.5</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>SECTION NO. 16</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1000.21</td>
<td>600.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1030.21</td>
<td>600.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.000</td>
<td>0.04</td>
<td>0.4</td>
<td>0.905</td>
<td>0.976</td>
<td>0.9925</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9963</td>
<td>0.9984</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>680.0</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>SECTION NO. 17</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Section</td>
<td>Height</td>
<td>Length</td>
<td>Width</td>
<td>Thickness</td>
<td>Water Depth</td>
<td>Water Height</td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>--------</td>
<td>--------</td>
<td>-------</td>
<td>-----------</td>
<td>-------------</td>
<td>--------------</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>0.000</td>
<td>0.02</td>
<td>0.23</td>
<td>0.87</td>
<td>0.963</td>
<td>0.983</td>
<td></td>
</tr>
<tr>
<td></td>
<td>690.0</td>
<td></td>
<td></td>
<td>2</td>
<td></td>
<td>SECTION NO. 18</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>0.000</td>
<td>0.03</td>
<td>0.3</td>
<td>0.925</td>
<td>0.97</td>
<td>0.985</td>
<td></td>
</tr>
<tr>
<td></td>
<td>700.0</td>
<td></td>
<td></td>
<td>2</td>
<td></td>
<td>SECTION NO. 19</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>0.000</td>
<td>0.029</td>
<td>0.24</td>
<td>0.82</td>
<td>0.95</td>
<td>0.979</td>
<td></td>
</tr>
<tr>
<td></td>
<td>710.0</td>
<td></td>
<td></td>
<td>2</td>
<td></td>
<td>SECTION NO. 20</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>0.000</td>
<td>0.024</td>
<td>0.4</td>
<td>0.925</td>
<td>0.97</td>
<td>0.987</td>
<td></td>
</tr>
<tr>
<td></td>
<td>720.0</td>
<td></td>
<td></td>
<td>2</td>
<td></td>
<td>SECTION NO. 21</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>0.000</td>
<td>0.016</td>
<td>0.24</td>
<td>0.75</td>
<td>0.915</td>
<td>0.965</td>
<td></td>
</tr>
<tr>
<td></td>
<td>731.2</td>
<td></td>
<td></td>
<td>2</td>
<td></td>
<td>SECTION NO. 22</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>0.000</td>
<td>0.028</td>
<td>0.42</td>
<td>0.93</td>
<td>0.973</td>
<td>0.988</td>
<td></td>
</tr>
<tr>
<td></td>
<td>740.0</td>
<td></td>
<td></td>
<td>2</td>
<td></td>
<td>SECTION NO. 23</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>0.000</td>
<td>0.032</td>
<td>0.35</td>
<td>0.717</td>
<td>0.9065</td>
<td>0.955</td>
<td></td>
</tr>
<tr>
<td></td>
<td>750.0</td>
<td></td>
<td></td>
<td>2</td>
<td></td>
<td>SECTION NO. 24</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>0.000</td>
<td>0.00</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td>SECTION NO. 25</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>0.000</td>
<td>0.0406</td>
<td>0.3786</td>
<td>0.7816</td>
<td>0.9054</td>
<td>0.9565</td>
<td></td>
</tr>
<tr>
<td></td>
<td>772.9</td>
<td></td>
<td></td>
<td>15</td>
<td></td>
<td>SECTION NO. 26</td>
<td></td>
</tr>
</tbody>
</table>

133
| 1107.30 | 0.0  | 0.0  | 0.0  |
| 1109.00 | 131.0 | 0.75 | 108.0 |
| 1111.00 | 595.0 | 2.17 | 274.0 |
| 1113.00 | 1249.0 | 3.02 | 414.0 |
| 1115.00 | 2483.0 | 2.8  | 867.0 |
| 1117.00 | 4611.0 | 3.54 | 1304.0 |
| 1119.00 | 7706.0 | 4.31 | 1790.0 |
| 1121.00 | 11530.0 | 5.62 | 2050.0 |
| 1123.00 | 16190.0 | 6.17 | 2622.0 |
| 1125.00 | 22921.0 | 5.52 | 4153.0 |
| 1127.00 | 31417.0 | 7.25 | 4355.0 |
| 1129.00 | 40359.0 | 6.64 | 4672.0 |
| 1131.00 | 50053.0 | 10.25 | 4862.0 |
| 1133.00 | 59820.0 | 12.25 | 4884.0 |
| 1135.00 | 69591.0 | 14.24 | 4867.0 |
| 0.000  | 0.0501 | 0.4105 | 0.7807 | 0.9043 | 0.9603 |
| 0.9806 | 0.9995 | 1.0000 | |
| 782.10 | 16 | | |

**SECTION NO. 27**

| 1110.90 | 0.0  | 0.0  | 0.0  |
| 1112.00 | 30.0 | 0.56 | 34.0 |
| 1114.00 | 218.0 | 1.03 | 134.0 |
| 1116.00 | 550.0 | 2.99 | 184.0 |
| 1118.00 | 942.0 | 4.51 | 209.0 |
| 1120.00 | 1398.0 | 5.41 | 256.0 |
| 1122.00 | 1968.0 | 6.57 | 295.0 |
| 1124.00 | 2586.0 | 8.11 | 319.0 |
| 1126.00 | 3242.0 | 9.59 | 388.0 |
| 1128.00 | 3991.0 | 8.48 | 471.0 |
| 1130.00 | 5277.0 | 5.65 | 902.0 |
| 1132.00 | 7974.0 | 3.41 | 2335.0 |
| 1134.00 | 15095.0 | 3.44 | 4619.0 |
| 1136.00 | 25240.0 | 5.35 | 4718.0 |
| 1138.00 | 34662.0 | 7.34 | 4723.0 |
| 1140.00 | 76023.0 | 11.51 | 5006.0 |
| 0.000  | 0.0574 | 0.4348 | 0.7906 | 0.9034 | 0.9632 |
| 0.984 | 0.9993 | 1.0000 | |
| 792.10 | 16 | | |

**SECTION NO. 28**

| 1123.00 | 0.0  | 0.0  | 0.0  |
| 1125.00 | 63.0 | 1.0  | 53.0 |
| 1127.00 | 234.0 | 2.44 | 96.0 |
| 1129.00 | 440.0 | 4.03 | 109.0 |
| 1131.00 | 671.0 | 5.5  | 122.0 |
| 1133.00 | 1082.0 | 4.86 | 222.0 |
| 1135.00 | 1544.0 | 6.07 | 234.0 |
| 1137.00 | 2342.0 | 5.31 | 441.0 |
| 1139.00 | 3250.0 | 5.69 | 573.0 |
| 1141.00 | 4734.0 | 5.18 | 924.0 |
| 1143.00 | 7081.0 | 4.59 | 1542.0 |
| 1145.00 | 10577.0 | 5.01 | 2113.0 |
| 1147.00 | 14811.0 | 6.70 | 2211.0 |
| 1149.00 | 19468.0 | 7.39 | 2467.0 |
| 1151.00 | 24839.0 | 8.30 | 2993.0 |
| 1153.00 | 38262.0 | 9.03 | 3894.0 |
| 0.000  | 0.0653 | 0.4609 | 0.7947 | 0.9024 | 0.9663 |
| 0.9878 | 0.9991 | 1.0000 | |
| 801.90 | 15 | | |

**SECTION NO. 29**

| 1134.50 | 0.0  | 0.0  | 0.0  |
| 1136.00 | 43.0 | 0.79 | 35.0 |
| 1138.00 | 205.0 | 1.63 | 122.0 |
| 1140.00 | 552.0 | 2.6 | 212.0 |
| 1142.00 | 1026.0 | 3.93 | 261.0 |
| 1144.00 | 1645.0 | 4.23 | 367.0 |
| 1146.00 | 2489.0 | 5.31 | 460.0 |
| 1148.00 | 3525.0 | 5.41 | 652.0 |
| 1150.00 | 4982.0 | 5.96 | 837.0 |
| 1152.00 | 6624.0 | 6.13 | 1113.0 |
| 1154.00 | 9955.0 | 4.99 | 1995.0 |
| 1156.00 | 14753.0 | 5.77 | 2558.0 |
| 1158.00 | 20036.0 | 7.2 | 2784.0 |
| 1160.00 | 26475.0 | 7.34 | 3609.0 |
| 1170.0 | 62585.0 | 17.34 | 3609.0 |

| 0.000 | 0.0731 | 0.4871 | 0.7989 | 0.9015 | 0.9694 | 311.0 |

| 1137.20 | 0.0 | 0.0 | 0.0 |
| 1139.00 | 9.0 | 0.4 | 23.0 |
| 1141.00 | 204.0 | 1.9 | 108.0 |
| 1143.00 | 465.0 | 3.07 | 151.0 |
| 1145.00 | 825.0 | 3.54 | 233.0 |
| 1147.00 | 1460.0 | 3.70 | 394.0 |
| 1149.00 | 2366.0 | 4.06 | 582.0 |
| 1151.00 | 3642.0 | 5.51 | 661.0 |
| 1153.00 | 5000.0 | 7.17 | 697.0 |
| 1155.00 | 6444.0 | 8.16 | 768.0 |
| 1157.00 | 8176.0 | 8.7 | 940.0 |
| 1159.00 | 10494.0 | 7.53 | 1393.0 |
| 1167.00 | 21633.0 | 15.53 | 1393.0 |
| 1180.0 | 39742.2 | 28.53 | 1393.0 |

| 0 | 7 | 1000.0 | 1 | 0 | 0 | 0 | 0.50 |
| 0 | 70.1 | 15000.0 | 192.0 | 60.0 | 504.0 | 498.0 | 62.0 | 147.0 | 2700.0 |

<p>| 138 | 36000.0 | 776.0 | 240.0 | 2016.0 | 1992.0 | 248.0 | 590.0 | 10800.0 |
| 120 | 36000.0 | 776.0 | 240.0 | 2016.0 | 1992.0 | 248.0 | 590.0 | 10800.0 |
| 121 | 36000.0 | 192.0 | 60.0 | 504.0 | 498.0 | 62.0 | 147.0 | 2700.0 |
| 240 | 36000.0 | 192.0 | 60.0 | 504.0 | 498.0 | 62.0 | 147.0 | 2700.0 |
| 242 | 70.1 | 15000.0 | 192.0 | 60.0 | 504.0 | 498.0 | 62.0 | 147.0 | 2700.0 |
| 360 | 70.1 | 15000.0 | 192.0 | 60.0 | 504.0 | 498.0 | 62.0 | 147.0 | 2700.0 |
| 361 | 36000.0 | 776.0 | 240.0 | 2016.0 | 1992.0 | 248.0 | 590.0 | 10800.0 |
| 480 | 36000.0 | 776.0 | 240.0 | 2016.0 | 1992.0 | 248.0 | 590.0 | 10800.0 |
| 481 | 36000.0 | 192.0 | 60.0 | 504.0 | 498.0 | 62.0 | 147.0 | 2700.0 |
| 600 | 36000.0 | 192.0 | 60.0 | 504.0 | 498.0 | 62.0 | 147.0 | 2700.0 |
| 601 | 70.1 | 15000.0 | 192.0 | 60.0 | 504.0 | 498.0 | 62.0 | 147.0 | 2700.0 |
| 720 | 70.1 | 15000.0 | 192.0 | 60.0 | 504.0 | 498.0 | 62.0 | 147.0 | 2700.0 |</p>
<table>
<thead>
<tr>
<th>Voltage (V)</th>
<th>Current (mA)</th>
<th>Resistance (Ω)</th>
<th>Power (W)</th>
<th>Input (W)</th>
<th>Output (W)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2520</td>
<td>70</td>
<td>15000</td>
<td>192</td>
<td>60</td>
<td>504</td>
</tr>
<tr>
<td>2521</td>
<td>38</td>
<td>36000</td>
<td>776</td>
<td>240</td>
<td>2016</td>
</tr>
<tr>
<td>2640</td>
<td>38</td>
<td>36000</td>
<td>776</td>
<td>240</td>
<td>2016</td>
</tr>
<tr>
<td>2641</td>
<td>38</td>
<td>36000</td>
<td>192</td>
<td>60</td>
<td>504</td>
</tr>
<tr>
<td>2760</td>
<td>38</td>
<td>36000</td>
<td>192</td>
<td>60</td>
<td>504</td>
</tr>
<tr>
<td>2761</td>
<td>70</td>
<td>15000</td>
<td>192</td>
<td>60</td>
<td>504</td>
</tr>
<tr>
<td>2880</td>
<td>70</td>
<td>15000</td>
<td>192</td>
<td>60</td>
<td>504</td>
</tr>
<tr>
<td>2881</td>
<td>38</td>
<td>36000</td>
<td>776</td>
<td>240</td>
<td>2016</td>
</tr>
<tr>
<td>3000</td>
<td>38</td>
<td>36000</td>
<td>776</td>
<td>240</td>
<td>2016</td>
</tr>
<tr>
<td>3001</td>
<td>38</td>
<td>36000</td>
<td>192</td>
<td>60</td>
<td>504</td>
</tr>
<tr>
<td>3120</td>
<td>38</td>
<td>36000</td>
<td>192</td>
<td>60</td>
<td>504</td>
</tr>
<tr>
<td>3121</td>
<td>70</td>
<td>15000</td>
<td>192</td>
<td>60</td>
<td>504</td>
</tr>
<tr>
<td>3240</td>
<td>70</td>
<td>15000</td>
<td>192</td>
<td>60</td>
<td>504</td>
</tr>
<tr>
<td>3241</td>
<td>38</td>
<td>36000</td>
<td>776</td>
<td>240</td>
<td>2016</td>
</tr>
<tr>
<td>3360</td>
<td>38</td>
<td>36000</td>
<td>776</td>
<td>240</td>
<td>2016</td>
</tr>
<tr>
<td>3361</td>
<td>38</td>
<td>36000</td>
<td>192</td>
<td>60</td>
<td>504</td>
</tr>
<tr>
<td>3480</td>
<td>38</td>
<td>36000</td>
<td>192</td>
<td>60</td>
<td>504</td>
</tr>
<tr>
<td>3481</td>
<td>70</td>
<td>15000</td>
<td>192</td>
<td>60</td>
<td>504</td>
</tr>
<tr>
<td>3600</td>
<td>70</td>
<td>15000</td>
<td>192</td>
<td>60</td>
<td>504</td>
</tr>
<tr>
<td>3601</td>
<td>38</td>
<td>36000</td>
<td>776</td>
<td>240</td>
<td>2016</td>
</tr>
<tr>
<td>3720</td>
<td>38</td>
<td>36000</td>
<td>776</td>
<td>240</td>
<td>2016</td>
</tr>
<tr>
<td>3721</td>
<td>38</td>
<td>36000</td>
<td>192</td>
<td>60</td>
<td>504</td>
</tr>
<tr>
<td>3840</td>
<td>38</td>
<td>36000</td>
<td>192</td>
<td>60</td>
<td>504</td>
</tr>
<tr>
<td>3841</td>
<td>70</td>
<td>15000</td>
<td>192</td>
<td>60</td>
<td>504</td>
</tr>
<tr>
<td>3960</td>
<td>70</td>
<td>15000</td>
<td>192</td>
<td>60</td>
<td>504</td>
</tr>
<tr>
<td>3961</td>
<td>38</td>
<td>36000</td>
<td>776</td>
<td>240</td>
<td>2016</td>
</tr>
<tr>
<td>4080</td>
<td>38</td>
<td>36000</td>
<td>776</td>
<td>240</td>
<td>2016</td>
</tr>
<tr>
<td>4081</td>
<td>38</td>
<td>36000</td>
<td>192</td>
<td>60</td>
<td>504</td>
</tr>
<tr>
<td>4200</td>
<td>38</td>
<td>36000</td>
<td>192</td>
<td>60</td>
<td>504</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>4201 70</td>
<td>15000.0</td>
<td>192.0</td>
<td>60.0</td>
<td>504.0</td>
<td>498.0</td>
</tr>
<tr>
<td>4320 70</td>
<td>15000.0</td>
<td>192.0</td>
<td>60.0</td>
<td>504.0</td>
<td>498.0</td>
</tr>
<tr>
<td>4321 38</td>
<td>36000.0</td>
<td>776.0</td>
<td>240.0</td>
<td>2016.0</td>
<td>1992.0</td>
</tr>
<tr>
<td>4440 38</td>
<td>36000.0</td>
<td>776.0</td>
<td>240.0</td>
<td>2016.0</td>
<td>1992.0</td>
</tr>
<tr>
<td>4441 38</td>
<td>36000.0</td>
<td>192.0</td>
<td>60.0</td>
<td>504.0</td>
<td>498.0</td>
</tr>
<tr>
<td>4560 38</td>
<td>36000.0</td>
<td>192.0</td>
<td>60.0</td>
<td>504.0</td>
<td>498.0</td>
</tr>
<tr>
<td>4561 70</td>
<td>15000.0</td>
<td>192.0</td>
<td>60.0</td>
<td>504.0</td>
<td>498.0</td>
</tr>
<tr>
<td>4680 70</td>
<td>15000.0</td>
<td>192.0</td>
<td>60.0</td>
<td>504.0</td>
<td>498.0</td>
</tr>
<tr>
<td>4681 38</td>
<td>36000.0</td>
<td>776.0</td>
<td>240.0</td>
<td>2016.0</td>
<td>1992.0</td>
</tr>
<tr>
<td>4800 38</td>
<td>36000.0</td>
<td>776.0</td>
<td>240.0</td>
<td>2016.0</td>
<td>1992.0</td>
</tr>
<tr>
<td>4801 38</td>
<td>36000.0</td>
<td>192.0</td>
<td>60.0</td>
<td>504.0</td>
<td>498.0</td>
</tr>
<tr>
<td>4920 38</td>
<td>36000.0</td>
<td>192.0</td>
<td>60.0</td>
<td>504.0</td>
<td>498.0</td>
</tr>
<tr>
<td>4921 70</td>
<td>15000.0</td>
<td>192.0</td>
<td>60.0</td>
<td>504.0</td>
<td>498.0</td>
</tr>
<tr>
<td>5040 70</td>
<td>15000.0</td>
<td>192.0</td>
<td>60.0</td>
<td>504.0</td>
<td>498.0</td>
</tr>
<tr>
<td>5041 38</td>
<td>36000.0</td>
<td>776.0</td>
<td>240.0</td>
<td>2016.0</td>
<td>1992.0</td>
</tr>
<tr>
<td>5160 38</td>
<td>36000.0</td>
<td>776.0</td>
<td>240.0</td>
<td>2016.0</td>
<td>1992.0</td>
</tr>
<tr>
<td>5161 38</td>
<td>36000.0</td>
<td>192.0</td>
<td>60.0</td>
<td>504.0</td>
<td>498.0</td>
</tr>
<tr>
<td>5280 38</td>
<td>36000.0</td>
<td>192.0</td>
<td>60.0</td>
<td>504.0</td>
<td>498.0</td>
</tr>
<tr>
<td>5281 70</td>
<td>15000.0</td>
<td>192.0</td>
<td>60.0</td>
<td>504.0</td>
<td>498.0</td>
</tr>
<tr>
<td>5400 70</td>
<td>15000.0</td>
<td>192.0</td>
<td>60.0</td>
<td>504.0</td>
<td>498.0</td>
</tr>
<tr>
<td>5401 38</td>
<td>36000.0</td>
<td>776.0</td>
<td>240.0</td>
<td>2016.0</td>
<td>1992.0</td>
</tr>
<tr>
<td>5520 38</td>
<td>36000.0</td>
<td>776.0</td>
<td>240.0</td>
<td>2016.0</td>
<td>1992.0</td>
</tr>
<tr>
<td>5521 38</td>
<td>36000.0</td>
<td>192.0</td>
<td>60.0</td>
<td>504.0</td>
<td>498.0</td>
</tr>
<tr>
<td>5640 38</td>
<td>36000.0</td>
<td>192.0</td>
<td>60.0</td>
<td>504.0</td>
<td>498.0</td>
</tr>
<tr>
<td>5641 70</td>
<td>15000.0</td>
<td>192.0</td>
<td>60.0</td>
<td>504.0</td>
<td>498.0</td>
</tr>
<tr>
<td>5760 70</td>
<td>15000.0</td>
<td>192.0</td>
<td>60.0</td>
<td>504.0</td>
<td>498.0</td>
</tr>
<tr>
<td>5761 38</td>
<td>36000.0</td>
<td>776.0</td>
<td>240.0</td>
<td>2016.0</td>
<td>1992.0</td>
</tr>
<tr>
<td>5880 38</td>
<td>36000.0</td>
<td>776.0</td>
<td>240.0</td>
<td>2016.0</td>
<td>1992.0</td>
</tr>
<tr>
<td>5881 38</td>
<td>36000.0</td>
<td>192.0</td>
<td>60.0</td>
<td>504.0</td>
<td>498.0</td>
</tr>
</tbody>
</table>