3 NUMERICAL APPROACHES

In this chapter, we are to present the numerical approaches employed to solve the governing equations of flow and transport given in the previous section. In our model, transport is assumed not to influence flow. Three time scales are considered in the model. They are (1) for three-dimensional subsurface flow, (2) for three-dimensional subsurface transport and two-dimensional overland flow/transport, and (3) for one-dimensional river/stream/canal flow/transport. In general, a three-dimensional flow time step may include several two-dimensional flow time steps and a two-dimensional flow time step can cover many one-dimensional flow time steps. The time scale for three-dimensional subsurface transport is set to be the same as that for two-dimensional overland flow/transport because kinetic chemical reactions are taken into account. During each three-dimensional flow time step, we solve three-dimensional subsurface flow by employing the updated two-dimensional flow conditions to achieve the surface/subsurface interface boundary conditions and determine the infiltration/seepage for two-dimensional flow computation included in this three-dimensional flow time step. During each two-dimensional flow time step, we first solve three-dimensional reactive chemical transport with the updated two-dimensional transport result (i.e., at the previous time) used for implementing variable boundary conditions on the interface boundary and determine the dissolve chemical flux through the surface/subsurface interface. This flux is actually the source/sink to two-dimensional dissolve chemical transport through infiltration/seepage. Then we solve two-dimensional flow equations to determine the water stage/depth and velocity of overland flow. Finally, we solve two-dimensional reactive chemical transport equations for the distribution of dissolved chemicals, sediments, and particulate chemicals. Within a one-dimensional flow time step, the river/stream flow equations are solved first and the one-dimensional transport equations are solved by using the newly-computed flow results. The interaction between one-dimensional river/stream and two-dimenional overland flow/transport is taken into account by using the updated computational results. Depth or stage difference-dependent fluxes are employed to determine the flow through this one-dimensional/two-dimensional interface.

3.1 Solving One-Dimensional River/Stream/Canal Network Flow Equations

As mentioned earlier in this report, we desire to implement a hybrid model to accurately simulate surface water flow under a wide range of physical conditions though it is still under investigation and further study is required. In our investigation to date, we would apply the hybrid Lagrangian-Eulerian finite element method to solve dynamical wave models, the hybrid Lagrangian-Eulerian or conventional finite element method to solve diffusion wave models, and the semi-Lagrangian method for kinematic wave models. In this and the next subsections, we will present the numerical approaches used in the method of characteristics and the Lagrangian approach for solving the one-dimensional river/stream/canal flow and two-dimensional overland flow equations, respectively. In either approach, the Picard method is employed to deal with the nonlinearity.

3.1.1 The Lagrangian-Eulerian Finite Element Method for Dynamic Wave

Substituting Equations (2.1.10) through (2.1.12) into Equations (2.1.19) and (2.1.20) and rearranging the resulting equations, we obtain

$$\frac{D_{V+c}(V+\omega)}{D\tau} = D - K_+ V + S_+ \qquad \textbf{(3.1.1)}$$

$$\frac{D_{V-c}(V-\omega)}{D\tau} = D - K_- V + S_- \qquad \textbf{(3.1.2)}$$

in which

$$D = \frac{1}{A}\frac{\partial}{\partial x}\left(\varepsilon A \frac{\partial V}{\partial x}\right); K_+ = \frac{g}{Bc}\frac{\partial A^{\#}}{\partial x} + \frac{(S_S + S_R - S_E + S_I + S_1 + S_2)}{A} + \frac{\kappa P V}{A} \qquad \textbf{(3.1.3)}$$

$$\begin{aligned} S_+ = \frac{g}{Bc}(S_S + S_R - S_E + S_I + S_1 + S_2) - g\frac{\partial Z_o}{\partial x} - \frac{gh}{c\rho}\frac{\partial \Delta\rho}{\partial x} \\ + \frac{(M_S + M_R - M_E + M_I + M_1 + M_2)}{A} + \frac{B\tau^s}{\rho A} \end{aligned} \qquad \textbf{(3.1.4)}$$

$$K_- = \frac{g}{Bc}\frac{\partial A^{\#}}{\partial x} + \frac{(S_S + S_R - S_E + S_I + S_1 + S_2)}{A} + \frac{\kappa P V}{A} \qquad \textbf{(3.1.5)}$$

$$\begin{aligned} S_- = -\frac{g}{Bc}(S_S + S_R - S_E + S_I + S_1 + S_2) - g\frac{\partial Z_o}{\partial x} - \frac{gh}{c\rho}\frac{\partial \Delta\rho}{\partial x} \\ + \frac{M_S + M_R - M_E + M_I + M_1 + M_2}{A} + \frac{B\tau^s}{\rho A} \end{aligned} \qquad \textbf{(3.1.6)}$$

where D is the diffusive transport of waves, K_+ is the decay coefficient of the positive gravity wave, S_+ is the source/sink of the positive wave, K_- is the decay coefficient of the negative gravity wave, and S_- is the source/sink of the negative wave.

Integrating Equations (3.1.1) and (3.1.2) along their respective characteristic lines from x_i at new time-level to x_{i1}^* and x_{i2}^* (Fig. 3.1-1), we obtain

$$\begin{aligned} \frac{(V_i + \omega_i) - (V_{il}^* + \omega_{il}^*)}{\Delta\tau_l} = \frac{1}{2}(D_i + D_{il}^*) - \frac{1}{2}\left((K_+)_i V_i + (K_+)_{il}^* V_{il}^*\right) \\ + \frac{1}{2}\left((S_+)_i + (S_+)_{il}^*\right), \quad I \in N \end{aligned} \qquad \textbf{(3.1.7)}$$

$$\begin{aligned} \frac{(V_i - \omega_i) - (V_{i2}^* - \omega_{i2}^*)}{\Delta\tau_l} = \frac{1}{2}(D_i + D_{i2}^*) - \frac{1}{2}\left((K_-)_i V_i + (K_-)_{i2}^* V_{i2}^*\right) \\ + \frac{1}{2}\left((S_-)_i + (S_-)_{i2}^*\right), \quad I \in N \end{aligned} \qquad \textbf{(3.1.8)}$$

where (referring to Figure 3.1-1) V_i, ω_i are the values of V and ω at x_i (x_i = coordinate of node i) at new time level; V_{i1}^* and ω_{i1}^* are the values of V and ω point x_{i1}^* (where x_{i1}^* is the location of a fictitious particle backward tracked from x_i along the first characteristics); $\Delta\tau_1$ is the time determined by backward tracking along the first characteristic; D_i is the value of D at node i at new time level; D_{i1}^* is the value of D at point x_{i1}^*; $(K_+)_i$ and $(S_+)_i$ are the values of K_+ and S_+, respectively at node i

at new time level; $(K_+)_{i1}^*$ and $(S_+)_{i1}^*$ are the values of K_+ and S_+, respectively at node x_{i1}^* ; N is the number of nodes; V_{i2}^* and ω_{i2}^* are the values of V and ω point x_{i2}^* (where x_{i2}^* is the location of a fictitious particle backward tracked from x_i along the second characteristics); $\Delta\tau_2$ is the time determined by backward tracking along the second characteristic; D_{i2}^* is the value of D at point x_{i2}^*; $(K_-)_i$ and $(S_-)_i$ are the values of K_- and S_-, respectively at node i at new time level; and $(K)_{i2}^*$ and $(S_-)_{i2}^*$ are the values of K_- and S_-, respectively at node x_{i2}^*.

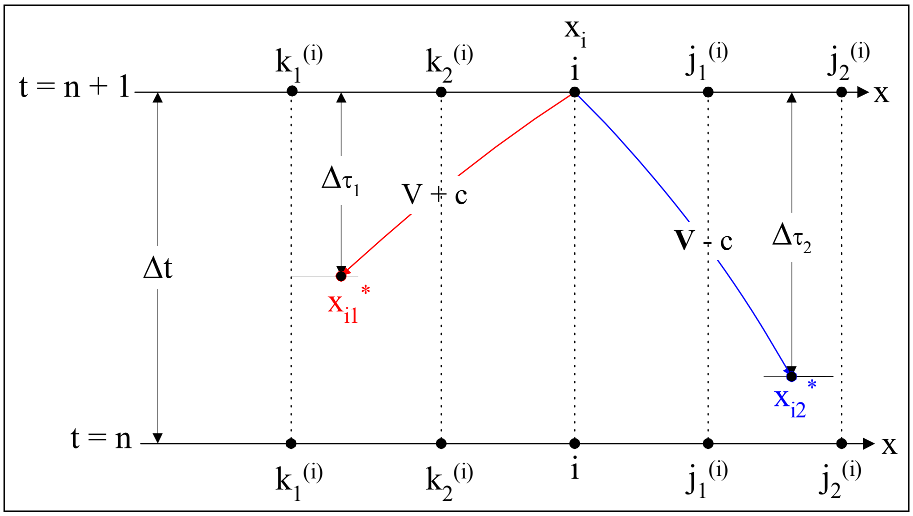

Fig. 3.1-1. Backward Tracking along Characteristics in One Dimension.

In Equations (3.1.7) and (3.1.8), the primitive variables at the backward tracked location are interpolated with those at the global nodes at both new time and old time as

$$V_{il}^* = a_{l(i)} V_{k_l^{(i)}}^{(n)} + a_{2(i)} V_{k_2^{(i)}}^{(n)} + a_{3(i)} V_{k_l^{(i)}} + a_{4(i)} V_{k_2^{(i)}} \qquad \textbf{(3.1.9)}$$

$$\omega_{il}^* = a_{l(i)} \omega_{k_l^{(i)}}^{(n)} + a_{2(i)} \omega_{k_2^{(i)}}^{(n)} + a_{3(i)} \omega_{k_l^{(i)}} + a_{4(i)} \omega_{k_2^{(i)}} \qquad \textbf{(3.1.10)}$$

$$V_{i2}^* = b_{l(i)} V_{j_l^{(i)}}^{(n)} + b_{2(i)} V_{j_2^{(i)}}^{(n)} + b_{3(i)} V_{j_l^{(i)}} + b_{4(i)} V_{j_2^{(i)}} \qquad \textbf{(3.1.11)}$$

$$\omega_{i2}^* = b_l \omega_{j_l^{(i)}}^{(n)} + b_2 \omega_{j_2^{(i)}}^{(n)} + b_3 \omega_{j_l^{(i)}} + b_4 \omega_{j_2^{(i)}} \qquad \textbf{(3.1.12)}$$

in which the superscript (n) denotes time level (n); $k_1^{(i)}$ and $k_2^{(i)}$ are the two nodes of the element in which the backward tracking from node i, along the first characteristic, stops; $j_1^{(i)}$ and $j_2^{(i)}$ are the two nodes of the element in which the backward tracking from node i, along the second characteristic, stops; $a_{1(i)}$, $a_{2(i)}$, $a_{3(i)}$, $a_{4(i)}$, $b_{1(i)}$, $b_{2(i)}$, $b_{3(i)}$, and $b_{4(i)}$ are the interpolation parameters associated with the backtracking of the i-th node, all in the range of [0,1]. It should be noted that we may use two given parameters to determine where to stop in the backward tracking: one is for controlling tracking time and the other one is for controlling tracking distance. After the primitive variables at the backward tracked points are interpolated, all other parameters (such as the decay coefficients and source/sink terms) are functions of these variables and can be calculated.

To compute the eddy diffusion terms D_i, we rewrite the first equation in Equation (3.1.3) as

$$AD = \frac{\partial}{\partial x}\left(A\varepsilon \frac{\partial V}{\partial x} \right) \quad \textbf{(3.1.13)}$$

in which the momentum flux due to turbulence is modeled with the eddy diffusion hypothesis. Applying the Galerkin finite element method to Equation (3.1.13), we obtain the following matrix equation for D as

$$[a]\{D\} + [b]\{V\} = \{F\} \quad \textbf{(3.1.14)}$$

in which

$$\{D\} = \{D_1 \quad D_2 \quad D_3 \quad . \quad . \quad D_i \quad . \quad . \quad D_N\}^T \quad \textbf{(3.1.15)}$$

$$\{V\} = \{V_1 \quad V_2 \quad V_3 \quad . \quad . \quad V_i \quad . \quad . \quad V_N\}^T \quad \textbf{(3.1.16)}$$

$$\{F\} = \{F_1 \quad F_2 \quad F_3 \quad . \quad . \quad F_i \quad . \quad . \quad F_N\}^T \quad \textbf{(3.1.17)}$$

$$a_{ij} = \int_{x_l}^{X_N} N_i A N_j dx, \quad b_{ij} = \int_{x_l}^{X_N} \frac{dN_i}{dx} A\varepsilon \frac{dN_j}{dx} dx, \quad F_i = n\mathrm{N}_\mathrm{i} A\varepsilon \frac{\partial V}{\partial x} \quad \textbf{(3.1.18)}$$

where N_i and N_j, functions of x, are the base functions of nodes at x_i and x_j, respectively.

Lumping the matrix $[a]$, we can solve Eq. (3.1.14) for D_i as follows

$$D_i = \frac{1}{a_{ii}} F_i - \frac{1}{a_{ii}} \sum_j b_{ij} V_j \quad \textbf{(3.1.19)}$$

Following the identical procedure that leads Eq. (3.1.13) to Eq. (3.1.19), we have

$$D_i^{(n)} = \frac{1}{a_{ii}^{(n)}} F_i^{(n)} - \frac{1}{a_{ii}^{(n)}} \sum_j b_{ij}^{(n)} V_j^{(n)} \quad \textbf{(3.1.20)}$$

where $\{F^{(n)}\}$, $\{a^{(n)}\}$ and $\{b^{(n)}\}$, respectively, are defined similar to $\{F\}$, $\{a\}$ and $\{b\}$, respectively. Similar to Eqs. (3.1.9) and (3.1.10), D_{i1}^* and D_{i2}^* at the backward tracked location are interpolated with $\{D\}$ and $\{D^{(n)}\}$ as

$$D_{il}^* = a_{l(i)} D_{k_l^{(i)}}^{(n)} + a_{(i)} D_{k_2^{(i)}}^{(n)} + a_{3(i)} D_{k_{l(}^{(i)}} + a_{4(i)} D_{k_2^{(i)}} \quad \textbf{(3.1.21)}$$

and

$$D_{i2}^* = b_{l(i)} D_{k_l^{(i)}}^{(n)} + b_{(i)} D_{k_2^{(i)}}^{(n)} + b_{3(i)} D_{k_l^{(i)}} + b_{4(i)} D_{k_2^{(i)}} \quad \textbf{(3.1.22)}$$

Substituting Equations (3.1.9) through (3.1.12) and Equations (3.1.19) through (3.1.22) into Equations (3.1.7) and (3.1.8) and implementing boundary conditions given Section 2.1.1, we obtain a system of $2N$ simultaneous algebraic equations for the $2N$ unknowns (V_i for $i = 1, 2, .., N$ and ω_i for

$i = 1, 2, .., N$). If the eddy diffusion terms are not included and the backward tracking is performed to reach the time level n (Fig. 3.1-2), then Eqs. (3.1.7) and (3.1.8) are reduced to a set of N decoupled pairs of equations as

$$a_{1l} V_i + a_{12}\omega_i = b_1 \quad and \quad a_{21} V_i - a_{22}\omega_i = b_2, \ \ i \in N \qquad \textbf{(3.1.23)}$$

$$a_{11} = 1 + \frac{\Delta\tau_1}{2}(K_+)_i, \ \ a_{12} = 1, \ \ b_1 = \left(1 - \frac{\Delta\tau_1}{2}(K_+)^*_{il}\right)V^*_{il} + \omega^*_{il} + \frac{\Delta\tau_1}{2}\left((S_+)_i + (S_+)^*_{il}\right),$$
$$a_{21} = 1 + \frac{\Delta\tau 2}{2}(K_-)_i, \ \ a_{22} = 1, \ \ b_2 = \left(1 - \frac{\Delta\tau_2}{2}(K_-)^*_{i2}\right)V^*_{i2} + \omega^*_{i2} + \frac{\Delta\tau_2}{2}\left((S_-)_i + (S_-)^*_{i2}\right), \qquad \textbf{(3.1.24)}$$

Equation (3.1.23)) is applied to all interior nodes without having to make any modification. On a boundary point, there are several possibilities: (1) both equations in Eq. (3.1.23) are replaced with two boundary equations, (2) one of the two equations is replaced with a boundary condition equation while the other remains unchanged, and (3) both equations stay valid. These conditions are addressed below.

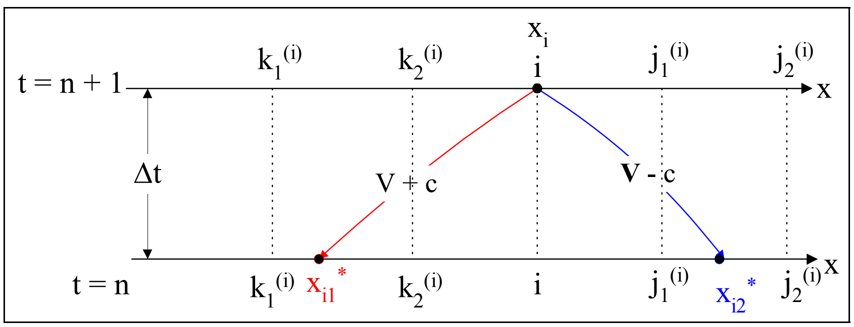

Fig. 3.1-2. Backward Tracking along Characteristics to the Toot in One Dimension.

Open upstream boundary condition:

If the flow is supercritical, Eq. (3.1.23) is replaced with

$$V_i A_i = Q_{up} \quad and \quad V_i^{\,2} A_i + g(h_c)_i A_i = M_{up} \qquad \textbf{(3.1.25)}$$

where V_i the cross-sectionally averaged velocity at node i, A_i is the cross-sectional area at node i, Q_{up} is the flow rate of the incoming fluid from the upstream, $(h_c)_i$ is the water depth to the centroid of the cross-sectional area at node i, and M_{up} is the momentum-impulse of the incoming fluid from the upstream. It should be noted that both the water depth and velocity in the upstream must be measured to provide values of Q_{up} and M_{up}. Equation (3.1.25) provides two equations for the solution of V_i and h_i. If the flow is critical, Eq. (3.1.23) for the boundary point i is replaced with

$$V_i A_i = Q_{up} \quad and \quad \frac{B_i Q_i^{\,2}}{g A_i^{\,3}} = 1 \qquad \textbf{(3.1.26)}$$

where B_i is the top width of the cross-section at node i. Equation (3.1.26) provides two equations to solve for V_i and h_i. If the flow is subcritical, Eq. (3.1.23) is replaced with

$$a_{11}V_i + a_{12}\omega_i = b_l \quad and \quad V_i A_i = Q_{up} \tag{3.1.27}$$

which is solved for V_i and h_i.

Open downstream boundary condition:

If the flow is supercritical, Eq. (3.1.23) is used to solve for V_i and h_i on node i. If the flow is critical, the following equation

$$a_{11}V_i + a_{12}\omega_i = b_1 \quad and \quad \frac{B_i Q_i^2}{g A_i^3} = 1 \tag{3.1.28}$$

is used to solve for V_i and h_i. If the flow is subcritical, the following equation is used to solve for V_i and h_i

$$a_{11}V_i + a_{12}\omega_i = b_1 \quad and \quad V_i A_i = Q_{dn}(h) \quad or \quad h_i = h_{dn}(t) \tag{3.1.29}$$

where $Q_{dn}(h)$, a function of h, is the rating curve function for the downstream boundary and $h_{dn}(t)$, a function of t, is the water depth at the downstream boundary. The adaption of Eq. (3.1.29) depends on the physical configuration at the boundary.

Closed upstream boundary condition:

If the flow is supercritical or critical, Eq. (3.1.23) is replaced with $V_i = 0$ and $h_i = 0$. If the flow is subcritical, $V_i = 0$ and the second equation in Eq. (3.1.23) is used to calculate h_i.

Closed downstream boundary conditions:

At the closed downstream boundary, physical condition dictates that the velocity at the boundary is zero. Therefore, supercritical flow cannot occur because c is greater or equal to zero. For critical flow, $V_i = 0$ and $h_i = 0$ at the closed boundary point x_i. For the subcritical flow, $V_i = 0$ and the first equation in Eq. (3.1.23) is used to calculate h_i.

Natural internal boundary condition at junctions:

For example, consider the junction node J joined by three reaches (Fig. 3.1-3), we have one unknown: the water surface elevation or the stage, H_J. The governing equation for this junction is

$$\frac{d\mathcal{V}_J}{dh_J}\frac{dh_J}{dt} = \sum_{I=1}^{I=3} Q_{IJ} = \sum_{I=1}^{I=3} V_{IJ} A_{IJ} \tag{3.1.30}$$

for the case when the storage effect of the junction is accounted for, or

$$\sum_{I=1}^{I=3} Q_{IJ} = \sum_{I=1}^{I=3} V_{IJ} A_{IJ} = 0 \tag{3.1.31}$$

for the case when the storage effect of the junction is small.

For the node IJ, we need to set up two equations for V_{IJ} and h_{IJ}. Let us say that node IJ is a downstream point if the flow is from the node IJ toward the junction J. On the other hand, we say that the node IJ is an upstream point if the flow is from the junction J toward the node IJ. Now we can set up two equations for each node IJ. This is demonstrated as follows.

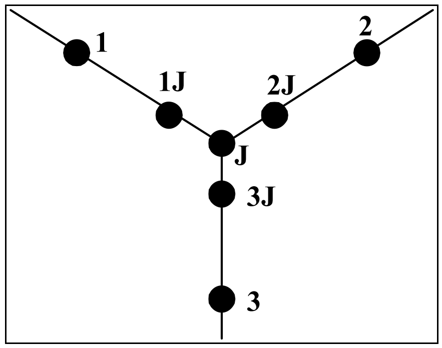

Fig. 3.1-3. A Three-Reach Junction

If IJ is a downstream point, we have three cases to consider:

(1). Subcritical flow –

$$a_{11}V_{IJ} + a_{12}\omega_{IJ} = b_1 \quad and \quad \frac{V_{IJ}^2}{2g} + h_{IJ} + Z_{oIJ} = H_J \tag{3.1.32}$$

(2). Supercritical flow –

$$a_{11}V_{IJ} + a_{12}\omega_{IJ} = b_1 \quad and \quad a_{21}V_{IJ} - a_{22}\omega_{IJ} = b_2 \tag{3.1.33}$$

(3). Critical flow –

$$a_{11}V_{IJ} + a_{12}\omega_{IJ} = b_1 \quad and \quad \frac{Q_{IJ}^2 B_{IJ}}{gA_{IJ}^3} = 1 \tag{3.1.34}$$

If IJ is an upstream point, we have three cases to consider:

(1) Subcritical flow -

$$\frac{V_{IJ}^2}{2g} + h_{IJ} + Z_{oIJ} = H_J \quad and \quad a_{21}V_{IJ} - a_{22}\omega_{IJ} = b_2 \tag{3.1.35}$$

(2). Supercritical flow –

$$\frac{V_{IJ}^{2}}{2g} + h_{IJ} + Z_{oIJ} = H_J \quad and \quad \frac{Q_{IJ}^{2} B_{IJ}}{g A_{IJ}^{3}} = 1 \tag{3.1.36}$$

(3). Critical flow –

$$\frac{V_{IJ}^{2}}{2g} + h_{IJ} + Z_{oIJ} = H_J \quad and \quad \frac{Q_{IJ}^{2} B_{IJ}}{g A_{IJ}^{3}} = 1 \tag{3.1.37}$$

Equation (3.1.30) or (3.1.31) and for I =1, 2, and 3, one of Eqs. (3.1.32) through (3.1.37) form 7 equations that can be solved for 7 unknowns V_{1J}, h_{1J}, V_{2J}, h_{2J}, V_{3J}, h_{3J}, and H_J. In theory, a substitution of the governing equations for the internal junction nodes into Eq. (3.1.30) or (3.1.31) eliminates all V_{IJ} and h_{IJ}, and the reduced Eq. (3.1.30) or (3.1.31) relates H_J to all unknowns at nodes other than that at node IJ. However, in practice, the 7 junction equations are solved simultaneously with all other discretized algebraic equations.

Controlled internal boundary condition at weirs:

For any weir (W), there are two river/stream/canal reaches connecting to it. The node *1W* located at the boundary between the 1^{th} reach and the W^{th} weir is termed the controlled internal boundary of the first reach while the node *2W* is called the controlled internal boundary of the second reach (Fig. 3.1-4). The specification of boundary conditions for the internal boundaries separated by a weir requires elaboration.

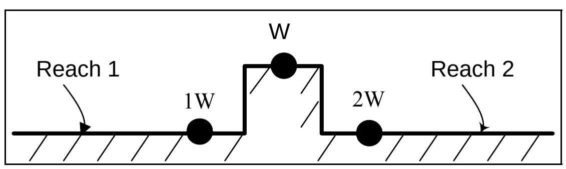

Fig. 3.1-4. A Flow-Control Weir

The flow configuration around the weir and its surrounding reaches may be very dynamic under transient flows. Both of the water stages at nodes *1W* and *2W* (H_{1W} and H_{2W}) may be below the weir, both may be above the weir, or one below the weir while the other is above the weir (Fig. 3.1-5). Governing equations of flow at internal boundary nodes *1W* and *2W* depend on the changing dynamics of water stages around the weir. When both stages H_{1W} and H_{2W} are below the height of the weir, the two reaches connecting the weir are decoupled. When at least one of the stages is above the weir, two reaches are either sequentially coupled or fully coupled via the weir. Here for sake of simplicity of discussions, we assume that the flow direction is from *Reach 1* to *Reach 2*. In other words, *Reach 1* is an upstream reach and *Reach 2* is a downstream reach. If the flow direction is reversed, we can have the boundary condition similarly prescribed.

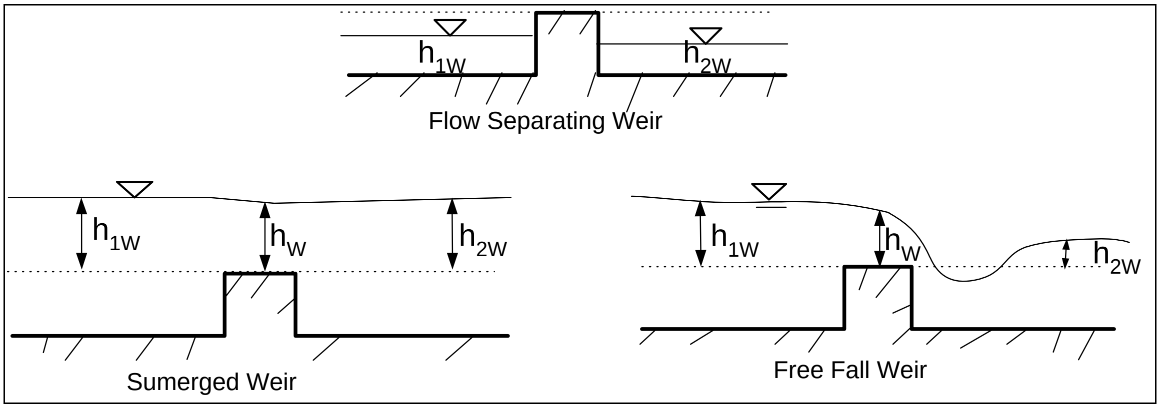

Fig. 3.1-5. Flow Configurations around a Weir.

There five unknowns, V_{1W} (velocity of the upstream reach node *1W*), h_{1W} (the water depth of the upstream node *1W*), Q_W (flow rate over the weir), V_{2W} (the velocity of the downstream reach node *2W*), and h_{2W} (the water depth of the downstream node *2W*); five equations must be set up for this weir complex consisting of a upstream reach node, a weir, and a downstream node. The governing equations for these five unknowns can be obtained depending on the flow conditions at the upstream and downstream reaches separated by a weir. The flow condition can be supercritical, critical, or subcritical at node *1W* and node *2W*. There are nine combinations. Five governing equations for each combination are given below.

Case 1: Supercritical flow at node 1W and supercritical flow at 2W (slowly varying flow)

$$a_{11}V_{1W} + a_{12}\omega_{1W} = b_1 \quad and \quad a_{21}V_{1W} - a_{22}\omega_{1W} = b_2 \tag{3.1.38}$$

$$Q_W = V_{1W}A_{1W}; \quad H_{1W} = h_{1W} + Z_{o1W} + \frac{V_{1W}^2}{2g}; \qquad M_{1W} = \rho\left(V_{1W}A_{1W}V_{1W} + gh_{1Wc}A_{1W}\right) \tag{3.1.39}$$

$$u_{2W}A_{2W} = Q_W \quad and \quad h_{2W} + Z_{o2W} + \frac{V_{2W}^2}{2g} + h_{LW} = H_{1W} \quad or$$
$$u_{2W}A_{2W} = Q_W \quad and \quad \rho\left(V_{2W}A_{2W}V_{2W} + gh_{2Wc}A_{2W}\right) + F_W = M_{1W} \tag{3.1.40}$$

where h_{LW} is the head loss between nodes *1W* and *2W* and F_W is the force exerted by the weir between nodes *1W* and *2W*. For this case, the computation is straightforward. First Eq. (3.1.38), which constitutes two equations for two unknowns V_{1W} and h_{1W}, is used to solve for these two unknowns. Then the flow rate through the weir, Q_W, and the momentum-impulse and energy line at point *1W*, M_{1W} and H_{1W}, are simply calculated with Eq. (3.1.39). Finally, either the first two equations or the last two equations in Eq. (3.1.40) constitute two equations for two unknowns V_{2W} and h_{2W}. These two unknowns are obtained by solving either first two equations or the last two equations in Eq. (3.1.40).

Case 2: Supercritical flow at node 1W and critical flow at 2W

$$a_{11}V_{1W} + a_{12}\omega_{1W} = b_1 \quad and \quad a_{21}V_{1W} - a_{22}\omega_{1W} = b_2 \tag{3.1.41}$$

$$Q_W = V_{1W}A_{1W} \tag{3.1.42}$$

$$V_{2W}A_{2W} = Q_W \quad and \quad \frac{Q_W^2 B_{2W}}{gA_{2W}^3} = 1 \tag{3.1.43}$$

For this case, the computation is straightforward. First Eq. (3.1.41), which constitutes two equations for two unknowns V_{1W} and h_{1W}, is used to solve for these two unknowns. Then the flow rate through the weir Q_W is simply calculated with Eq. (3.1.42). Finally, Equation (3.1.43) constitutes two equations for two unknowns V_{2W} and h_{2W}. These two unknowns are obtained by solving the two equations in Eq. (3.1.43).

Case 3: Supercritical flow at node *1W* and subcritical flow at *2W* (Hydraulic Jump)

$$a_{11}V_{1W} + a_{12}\omega_{1W} = b_1 \quad and \quad a_{21}V_{1W} - a_{22}\omega_{1W} = b_2 \tag{3.1.44}$$

$$Q_W = V_{1W}A_{1W} \tag{3.1.45}$$

$$a_{21}V_{2W} - a_{22}\omega_{2W} = b_2 \quad and \quad u_{2W}A_{2W} = Q_W \tag{3.1.46}$$

For this case, the computation is straightforward. First Eq. (3.1.44), which constitutes two equations for two unknowns V_{1W} and h_{1W}, is used to solve for these two unknowns. Then the flow rate through the weir Q_W is simply calculated with Eq. (3.1.45). Finally, Equation (3.1.46) constitutes two equations for two unknowns V_{2W} and h_{2W}. These two unknowns are obtained by solving the two equations in Eq. (3.1.46).

Case 4: Critical flow at node 1W and supercritical flow at 2W

$$a_{11}V_{1W} + a_{12}\omega_{1W} = b_1 \quad and \quad \frac{Q_{1W}^2 B_{1W}}{gA_{1W}^3} = 1, \tag{3.1.47}$$

$$Q_W = V_{1W}A_{1W};\ \ H_{1W} = h_{1W} + Z_{o1W} + \frac{V_{1W}^2}{2g};\ \ M_{1W} = \rho\left(V_{1W}A_{1W}V_{1W} + gh_{1Wc}A_{1W}\right) \tag{3.1.48}$$

$$\begin{aligned} &u_{2W}A_{2W} = Q_W \quad and \quad h_{2W} + Z_{o2W} + \frac{V_{2W}^2}{2g} + h_{LW} = H_{1W} \quad or \\ &u_{2W}A_{2W} = Q_W \quad and \quad \rho\left(V_{2W}A_{2W}V_{2W} + gh_{2Wc}A_{2W}\right) + F_w = M_{1W} \end{aligned} \tag{3.1.49}$$

For this case, the computation is straightforward. First Eq. (3.1.47), which constitutes two equations for two unknowns V_{1W} and h_{1W}, is used to solve for these two unknowns. Then the flow rate through the weir Q_W and the momentum-impulse and energy line at point *1W*, M_{1W} and H_{1W}, are simply calculated with Eq. (3.1.48). Finally, either the first two equations or the last two equations in Eq. (3.1.49) constitute two equations for two unknowns V_{2W} and h_{2W}. These two unknowns are obtained by solving either two equations or the last two equations in Eq. (3.1.49).

Case 5: Critical flow at node 1W and critical flow at 2W

$$a_{11}V_{1W} + a_{12}\omega_{1W} = b_1 \quad and \quad \frac{Q_{1W}^2 B_{1W}}{gA_{1W}^3} = 1, \qquad \textbf{(3.1.50)}$$

$$Q_W = V_{1W} A_{1W} \qquad \textbf{(3.1.51)}$$

$$V_{2W} A_{2W} = Q_W \quad and \quad \frac{Q_{2W}^2 B_{2W}}{gA_{2W}^3} = 1 \qquad \textbf{(3.1.52)}$$

For this case, the computation is straightforward. First Eq. (3.1.50), which constitutes two equations for two unknowns V_{1W} and h_{1W}, is used to solve for these two unknowns. Then the flow rate through the weir Q_W is simply calculated with Eq. (3.1.51). Finally, Equation (3.1.52) constitutes two equations for two unknowns V_{2W} and h_{2W}. These two unknowns are obtained by solving the two equations in Eq. (3.1.52).

Case 6: Critical flow at node 1W and subcritical flow at 2W (Hydraulic Jump)

$$a_{11}V_{1W} + a_{12}\omega_{1W} = b_1 \quad and \quad \frac{Q_{1W}^2 B_{1W}}{gA_{1W}^3} = 1, \qquad \textbf{(3.1.53)}$$

$$Q_W = V_{1W} A_{1W} \qquad \textbf{(3.1.54)}$$

$$a_{21}V_{2W} - a_{22}\omega_{2W} = b_2 \quad and \quad V_{2W} A_{2W} = Q_W \qquad \textbf{(3.1.55)}$$

For this case, the computation is straightforward. First Eq. (3.1.53), which constitutes two equations for two unknowns V_{1W} and h_{1W}, is used to solve for these two unknowns. Then the flow rate through the weir Q_W is simply calculated with Eq. (3.1.54). Finally, Equation (3.1.46) constitutes two equations for two unknowns V_{2W} and h_{2W}. These two unknowns are obtained by solving the two equations in Eq. (3.1.55).

Case 7: Subcritical flow at node 1W and Supercritical flow at 2W (Critical must occur at the weir)

$$\mathrm{a}_{11}\mathrm{V}_{1\mathrm{W}} + \mathrm{a}_{12}\omega_{1\mathrm{W}} = \mathrm{b}_1, \quad \mathrm{V}_{1\mathrm{W}}\mathrm{A}_{1\mathrm{W}} - \mathrm{Q}_\mathrm{W} = 0 \qquad \textbf{(3.1.56)}$$

$$\frac{Q_W^2 B_W}{gA_W^3} = 1, \; V_W A_W = Q_W, \; and \quad \begin{matrix} h_W + Z_{oW} + \dfrac{V_W^2}{2g} + h_{L1W} = h_{1W} + Z_{o1W} + \dfrac{V_{1W}^2}{2g} \\ or \\ \rho\left(V_W A_W V_W + g h_{Wc} A_W\right) + F_{1W} = \rho\left(V_{1W} A_{1W} V_{1W} + g h_{1Wc} A_{1W}\right) \end{matrix} \qquad \textbf{(3.1.57)}$$

$$u_{2W}A_{2W} - Q_w = 0 \;\; and \quad \begin{array}{c} h_{2W} + Z_{o2W} + \dfrac{V_{2W}^2}{2g} + h_{L2W} = h_W + Z_{oW} + \dfrac{V_{W_p}}{2g} \\ or \\ \rho\left(V_{2W}A_{2W}V_{2W} + gh_{2Wc}A_{2W}\right) + F_{2W} = \rho\left(V_W A_W V_W + gh_{Wc}A_W\right) \end{array} \qquad \textbf{(3.1.58)}$$

where h_{L1W} is the head loss between the weir and node $1W$, F_{1W} is the force exerted by the weir between the weir and node $1W$, h_{L2W} is the head loss between the weir and node $2W$, and F_{2W} is the force exerted by the weir between the weir and node $2W$. For this case, in addition to the five unknowns, V_{1W}, h_{1W}, Q_W, V_{2W}, and h_{2W}, two more unknowns, h_W and V_W, appear in Eqs. (3.1.56) through (3.1.58). These seven unknowns are obtained by solving seven simultaneous equations contained in Eqs. (3.1.56) through (3.1.58).

Case 8: Subcritical flow at node 1W and critical flow at 2W

$$a_{11}V_{1W} + a_{12}\omega_{1W} = b_1, \quad V_{1W}A_{1W} - Q_W = 0 \qquad \textbf{(3.1.59)}$$

$$V_{2W}A_{2W} - Q_W = 0, \quad \frac{Q_W^2 B_{2W}}{gA_{2W}^3} = 1, \;\; and \quad \begin{array}{c} h_{2W} + Z_{o2W} + \dfrac{V_{2W}^2}{2g} + h_{LW} = h_{1W} + Z_{o1W} + \dfrac{V_{1W}^2}{2g} \\ or \\ \rho\left(V_{2W}A_{2W}V_{2W} + gh_{2Wc}A_{2W}\right) + F_W = \rho\left(V_{1W}A_{1W}V_{1W} + gh_{1Wc}A_{1W}\right) \end{array} \qquad \textbf{(3.1.60}$$

For this case, five equations in Eqs. (3.1.59) and (3.1.60) are solved for the five unknowns, V_{1W}, h_{1W}, Q_W, V_{2W}, and h_{2W}.

Case 9: Subcritical flow at node 1W and Subcritical flow at 2W (slowly varying flow)

$$a_{11}V_{1W} + a_{12}\omega_{1W} = b_1, \quad V_{1W}A_{1W} - Q_W = 0 \qquad \textbf{(3.1.61)}$$

$$\begin{array}{l} a_{21}V_{2W} - a_{22}\omega_{2W} = b_2, \quad V_{2W}A_{2W} - Q_W = 0 \\ \quad and \\ h_{2W} + Z_{o2W} + \dfrac{V_{2W}^2}{2g} + h_{LW} = h_{1W} + Z_{o1W} + \dfrac{V_{1W}^2}{2g} \\ \qquad\qquad or \\ \rho\left(V_{2W}A_{2W}V_{2W} + gh_{2Wc}A_{2W}\right) + F_W = \rho\left(V_{1W}A_{1W}V_{1W} + gh_{1Wc}A_{1W}\right) \end{array} \qquad \textbf{(3.1.62)}$$

For this case, five equations in Eqs. (3.1.59) and (3.1.60) are solved for the five unknowns, V_{1W}, h_{1W}, Q_W, V_{2W}, and h_{2W}

Controlled internal boundary condition at Gates:

For any gate (G), there are two river/stream/canal reaches connecting to it. The node *1G* located at the boundary between the 1^{th} reach and the G^{th} gate is termed the controlled internal boundary of the first reach while the node *2G* is called the controlled internal boundary of the second reach (Fig. 3.1-6). The specification of boundary conditions for the internal boundaries separated by a gate can be made similar to that of a weir.

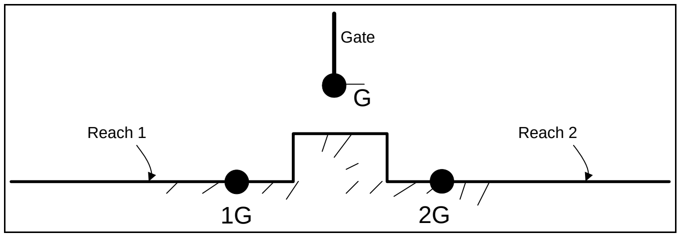

Fig. 3.1-6. A Flow-Control Gate.

The flow configuration around the gate and its surrounding reaches may be very dynamic under transient flows. Depending on the water stages at nodes *1G* and *2G* (H_{1G} and H_{2G}), we have several configurations (Fig. 3.1-7). Governing equations for flow at nodes *1G* and *2G* and through the gate depend on the changing dynamics of water stages around the gate. These equations can be obtained identical to those for a weir by changing the letter from *W* to *G*. Similar approaches can be used for culverts change the letter from *W* to *C* (for culverts). The only differences among various types of structures are the formulation of energy losses over the structures and/or the formulation of forces exerting on the fluids by the structures.

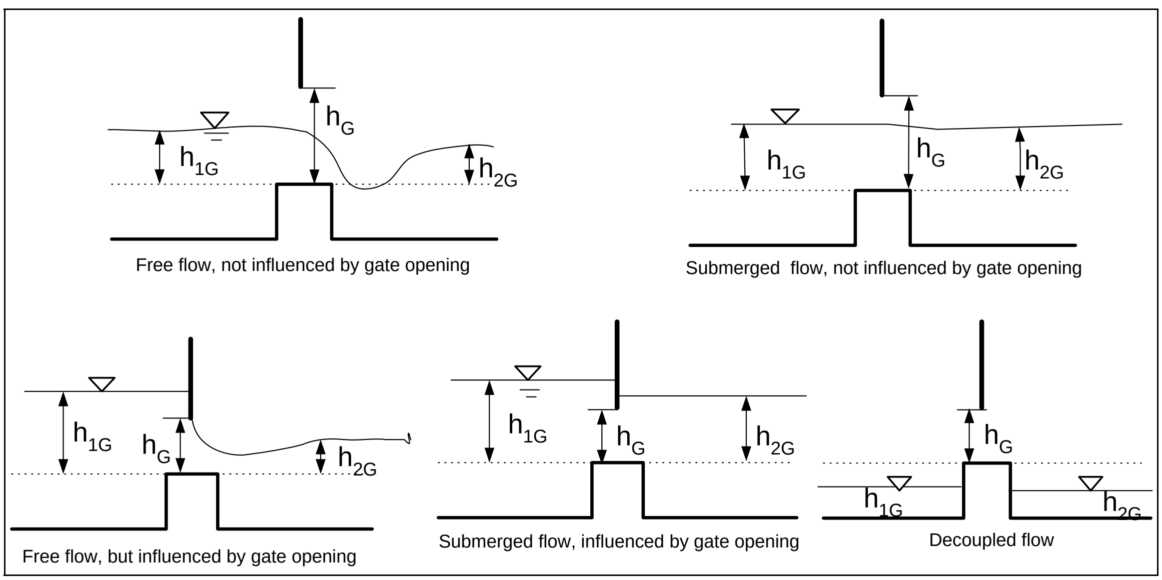

Fig. 3.1-7. Flow Configurations around a Gate.

3.1.2 Numerical Approximations of Diffusive Wave Approaches.

Two options are provided in this report to solve the diffusive wave flow equations. One is the finite element method and the other is the particle tracking method.

3.1.2.1 Galerkin Finite Element Method. Recall the diffusive wave is governed by Eq. (2.1.47) which is repeated here as

$$B\frac{\partial H}{\partial t}-\frac{\partial}{\partial x}\left(K\left[\frac{\partial H}{\partial x}+\frac{h}{c\rho}\frac{\partial \Delta\rho}{\partial x}-\frac{B\tau^s}{Ag\rho}\right]\right)=S_S+S_R-S_E+S_I+S_1+S_2 \quad \textbf{(3.1.63)}$$

Applying the Galerkin finite element method to Eq. (3.1.63), we obtain the following matrix equation.

$$[M]\frac{d\{H\}}{dt}+[S]\{H\}=\{Q_{\rho w}\}+\{Q_B\}+\{Q_S\}+\{Q_R\}-\{Q_E\}+\{Q_I\}+\{Q_1\}+\{Q_2\} \quad \textbf{(3.1.64)}$$

in which

$$M_{ij}=\int_{X_1}^{X_N}N_iBN_jdx,\quad S_{ij}=\int_{X_1}^{X_N}\frac{dN_i}{dx}K\frac{dN_j}{dx}dx,\quad Q_{\rho wi}=\int_{X_1}^{X_N}\frac{dN_i}{dx}K\left[\frac{h}{c\rho}\frac{\partial\Delta\rho}{\partial x}-\frac{B\tau^s}{Ag\rho}\right]$$
$$Q_i=nN_iK\left[\frac{\partial H}{\partial x}+\frac{h}{c\rho}\frac{\partial\Delta\rho}{\partial x}-\frac{B\tau^s}{Ag\rho}\right] \quad \textbf{(3.1.65)}$$

$$Q_{Si}=\int_{X_1}^{X_N}N_iS_Sdx,\quad Q_{Ri}=\int_{X_1}^{X_N}N_iS_Rdx,\quad Q_{Ei}=\int_{X_1}^{X_N}N_iS_Edx,$$
$$Q_{Ii}=\int_{X_1}^{X_N}N_iS_Idx,\quad Q_{1i}=\int_{X_1}^{X_N}N_iS_1dx,\quad Q_{2i}=\int_{X_1}^{X_N}N_iS_2dx, \quad \textbf{(3.1.66)}$$

where N_i and N_j are the base functions of nodes at x_i and x_j, respectively; n is the unit outward direction, n = 1 at a downstream point and n = -1 at an upstream point; [M] is the mass matrix, [S] is the stiff matrix, {H} is the solution vector of H, $\{Q_{\rho w}\}$ is the load vector due to density and wind stress effects, $\{Q_B\}$ is the flow rate through the boundary nodes of a river/stream/canal reach, $\{Q_S\}$ is the flow rate from artificial source/sink, $\{Q_R\}$ is the flow rate from rainfall, $\{Q_E\}$ is the flow rate due evapotranspiration, $\{Q_I\}$ is the flow rate to infiltration, $\{Q_1\}$ is the flow rate from overland flow via river bank *1*, and $\{Q_2\}$ is the flow rate from overland flow via river bank *2*. It should be noted that $\{Q_I\}$ is the interaction between the river/stream/canal reach and subsurface flows and $\{Q_1\}$ and $\{Q_2\}$ between the river/stream/canal (via bank 1 and bank 2) and overland flows.

Approximating the time derivative term in Eq. (3.1.64) with a time-weighted finite difference, we reduce the diffusive equation and its boundary conditions to the following matrix equation

$$[C]\{H\}=\{L\}+\{Q_B\}+\{Q_I\}+\{Q_1\}+\{Q_2\} \quad \textbf{(3.1.67)}$$

in which

$$[C]=\frac{[M]}{\Delta t}+\theta[S],\{L\}=\left(\frac{[M]}{\Delta t}-(1-\theta[S])\right)\{H^{(n)}\}+\{Q_{\rho w}\}+\{Q_S\}+\{Q_R\}-\{Q_E\} \quad \textbf{(3.1.68)}$$

where [C] is the coefficient matrix, {L} is the load vector from initial condition, density and wind effects, artificial sink/sources, rainfall, and evapotranspiration; Δt is the time step size; θ is the time weighting factor; and $\{H^{(n)}\}$ is the value of {H} at old time level n. The global and internal boundary (junctions, weirs, and gates) conditions must be used to provide $\{Q_B\}$ in Eq. (3.1.67). The interaction between the overland and river/stream/canal flows must be implemented to evaluate $\{Q_1\}$ and $\{Q_2\}$; and the interaction between the subsurface and river/stream/canal flows must be implemented to calculate $\{Q_I\}$. The interactions will be addressed in Section 3.4.

For a global boundary node *I*, the corresponding algebraic equation from Eq. (3.1.67) is

$$C_{I,I-1}H_{I-1} + C_{I,I}H_I = L_1 + Q_{BI} + Q_{II} + Q_{1I} + Q_{2I} \quad \textbf{(3.1.69)}$$

where (*I-1*) is the corresponding interior node of the node *I*. In the above equation there are two unknowns H_I and Q_{BI}; either H_I or Q_{BI}, or the relationship between H_I and Q_{BI} must be specified. The numerical implementation of these boundary conditions are described as follows.

Dirichlet-boundary condition: prescribed water depth or state

If H_I is given on the boundary node *I* (Dirichlet boundary condition), all coefficients ($C_{I,I-1}$, $C_{I,\mathrm{I}}$, $C_{I,I+1}$) and right-hand side (L_I, Q_{II}, Q_{1I}, Q_{2I}) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then an identity equation is created as

$$H_I = H_{Id}, \quad I \in N_D \quad \textbf{(3.1.70)}$$

where $\mathrm{H_{Id}}$ is the prescribed total head on the Dirichlet node *I* and N_D is the number of Dirichlet boundary nodes. This process is repeated for every Dirichlet nodes. Note it is unnecessary to modify other equations that involving these unknowns, which was done in the previous version. By not modifying other equations, the symmetrical property of the matrix is preserved, which makes the iterative solvers more robust. The final set of equations will consist of N_D identity equations and (N - N_D) finite element equations for *N* unknowns H_i*'s*. After H_i*'s* are obtained, Eq. (3.1.69) is then used to back calculate N_D Q_{BI}*'s*.

If a direct solver is used to solve the matrix equation, the above procedure will solve *N* H_i*'s* accurately except for roundoff errors. However, if an iterative solver is used, a stopping criteria must be strict enough so that the converged solution of *N* H_i*'s* are accurate enough to the exact solution. With such accurate H_i*'s*, then one can be sure that the back-calculated N_D Q_{BI}*'s* are accurate.

Flux boundary condition: prescribed flow rate

If Q_{BI} is given (flux boundary condition), all coefficients ($C_{I,I-1}$, $C_{I,I}$, $C_{I,I+1}$) and right-hand side (L_I, Q_{II}, Q_{1I}, Q_{2I}) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then Eq. (3.1.69) is modified to incorporate the boundary conditions and used to solve for H_I. The modification of Eq. (3.1.69) is straightforward. Because Q_{BI} is a known quantity, it contributes to the load on the right hand side. This type of boundary conditions is very easy to implement. After H_i*'s* are obtained, the original Eq. (3.1.69), which is stored in a temporary array, is used to back calculate N_C Q_{BI}*'s* on flux boundaries (where N_C is the number of flux boundary nodes). These back-calculated Q_{BI}*'s* should be theoretically identical to the input Q_{BI}*'s*. However, because of round-off errors (in the case of direct solvers) or because of stopping criteria (in the case of iterative solvers), the back-calculated Q_{BI}*'s* will be slightly different from the input Q_{BI}*'s*. If the differences between the two are significant, it is an indication that the solvers have not yielded accurate solutions.

Water depth-dependent boundary condition: prescribed rating curve

If the relationship is given between Q_{BI} and H_I (rating curve boundary condition), all coefficients ($C_{I,I-1}$, $C_{I,I}$, $C_{I,I+1}$) and right-hand side (L_I, Q_{II}, Q_{1I}, Q_{2I}) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then Eq. (3.1.69) is modified to incorporate the boundary conditions and used to solve for H_I. The rating-relationship is used to eliminate one of the unknowns, say Q_{BI}, and the modified Eq. (3.1.69) is used to solve for, say H_I. After H_I is solved, the original Eq. (3.1.69) (recall the original Eq. (3.1.69) must be and has been stored in a temporary array) is used to back-calculate Q_{BI}.

Junction boundary condition:

If the node *IJ* is an internal node that connects a junction *J*, then node *IJ* is treated as an internal boundary node. For example, consider three reaches with three internal nodes connecting to the junction *J* (Fig. 3.1-8). After applying the finite element method to Eq. (3.1.63), we have a total of (*1J* + *2J* + *3J*) algebraic equations. The algebraic equations for Nodes *1J*, *2J*, and *3J* can be written based on Eq. (3.1.69)

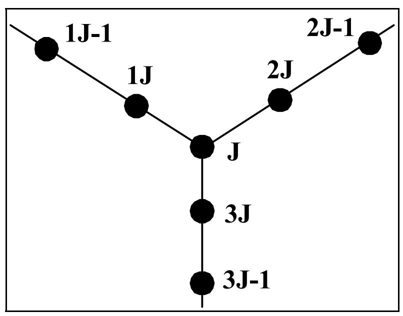

Fig. 3.1-8. A Three-Reach Junction

$$C^1_{1J,1J-1}H^1_{1J-1} + C^1_{1J,1J}H^1_{1J} = L^1_{1J} + Q^1_{1J} + Q^1_{I1J} + Q^1_{11J} + Q^1_{21J} \quad \textbf{(3.1.71)}$$

$$C^2_{2J,2J-1}H^2_{2J-1} + C^2_{2J,2J}H^2_{2J} = L^2_{2J} + Q^2_{2J} + Q^2_{I2J} + Q^2_{12J} + Q^2_{22J} \quad \textbf{(3.1.72)}$$

$$C^3_{3J,3J-1}H^3_{3J-1} + C^3_{3J,3J}H^3_{3J} = L^3_{3J} + Q^3_{3J} + Q^3_{I3J} + Q^3_{13J} + Q^3_{23J} \quad \textbf{(3.1.73)}$$

where the superscript denotes the reach number and subscript denotes local node number in a reach, for example, H_{1J}^1 denotes the total head at the *1J*-th node in Reach *1*. For a convenient discussion, let us associate each of the unknowns, H_1^1, …, H_{1J-1}^1 to each of the *1J-1* finite element equations in Reach *1*. Similarly, we associate each of the unknowns, H_1^2, .., H_{2J-2}^2 to each of the *2J-1* finite element equations in Reach *2* and each of the unknowns and H_1^3, .., H_{3J-1}^3 to each of the *3J-1* finite element equations in Reach *3*. The unknown, Q_{1J}^1, Q_{2J}^2, and Q_{3J}^3, are absent from these (*1J-1* + *2J-1* + *3J-1*) equations. In other words, we can say each equation governs one unknown. However, two unknowns, H_{1J}^1 and Q_{1J}^1, appear in Eq. (3.1.71). Similarly, Equation (3.1.72) has two unknowns, H_{2J}^2 and Q_{2J}^2, and Equation (3.1.73) has two unknowns, H_{3J}^3 and Q_{3J}^3. The number of unknowns, (1J + 2J + 3J) total heads and Q_{1J}^1, Q_{2J}^2, and Q_{3J}^3, is more than the number of equations, (*1J* + *2J* + *3J*) finite element equations. Three more governing equations must be set up, which can be obtained based on the continuity of energy lines. This is described as follows.

Assume the entrance loss to the junction and exit loss from the junction are negligible, we have the following three equations

$$H_{1j}^{1} + \frac{1}{2g}\left(\frac{Q_{1J}^{1}}{A_{1J}^{1}}\right)^2 = h_J + Z_{oJ} \tag{3.1.74}$$

$$H_{2j}^{2} + \frac{1}{2g}\left(\frac{Q_{2J}^{2}}{A_{2J}^{2}}\right)^2 = h_J + Z_{oJ} \tag{3.1.75}$$

$$H_{3j}^{3} + \frac{1}{2g}\left(\frac{Q_{3J}^{3}}{A_{3J}^{3}}\right)^2 = h_J + Z_{oJ} \tag{3.1.76}$$

where A_{1J}^{1}, A_{2J}^{2}, and A_{3J}^{3} are the cross-sectional area at *Nodes 1J* of *Reach 1*, *Node 2J* of *Reach 2*, and *Node 3J* of *Reach 3*, respectively; h_J is the water depth at the Junction J; and Z_{oJ} is the bottom elevation at the *Junction J*. It is noted that the second terms on the left hand side of Eqs. (3.1.74) through (3.1.76) are generally ignored in computation implementation to give more robust solutions.

The water depth at *Junction J* is not decoupled from river/stream/canal reaches. The water budget equation for the *Junction J* is

$$\frac{d\mathcal{V}_J}{dh_J}\frac{dh_J}{dt} = \sum_{i=1}^{i=3} Q_{iJ}^{i} \tag{3.1.77}$$

When $\frac{d\mathcal{V}_J}{dh_J}$ is small, the water budget Eq. (3.1.77) is not employed. Instead, the following equation, resulting from the requirement that the summation of flow rates is equal to zero, is used

$$\sum_{i=1}^{i=3} Q_{iJ}^{i} = 0 \tag{3.1.78}$$

Equations (3.1.71) through (3.1.76) and Eq. (3.1.77) or Eq. (3.1.78) constitute 7 equations for seven unknowns, A_{1J}^{1}, A_{2J}^{2}, A_{3J}^{3}, Q_{1J}^{1}, Q_{2J}^{2}, Q_{3J}^{3}, and h_J. If there are N_J junctions, there will be N_J blocks of seven equations. These N_J blocks of equations should be solved iteratively along with N_R block of finite element equations where N_R is the number of reaches. In other words, the whole system of algebraic equations can be solved with block iterations. Each block of equations can be solved directly. For example, each of N_R block of finite element equations can be solved with an efficient tri-diagonal matrix solver such as the Thomas algorithm. Each of the N_J block of seven equations can be solved with the Gaussian direct elimination with full pivoting.

Control Structure Boundary Condition:

The control structures may include weirs, gates, culverts, etc. For the two internal boundary nodes separated by a weir (Fig. 3.1-9), $Q_{1W} = Q_{2W} = Q_W$, where Q_W is given by

$$Q_W = C_W B_W h_{2W} \sqrt{h_{1W} - h_{2W}} \quad if \quad h_{1W} > h_{2W} > \frac{2}{3} h_{1W} \ (Submerged\ Weir) \qquad \textbf{(3.1.79)}$$

$$Q_W = \frac{2}{3\sqrt{3}} C_W B_W h_{1W} \sqrt{h_{1W}} \quad if \quad h_{2W} < \frac{2}{3} h_{1W} \ (Free\ Fall\ Weir) \qquad \textbf{(3.1.80)}$$

where C_W is the weir coefficient, B_W is the weir width [L]. The flow rate Q_W is equal to zero when both the upstream and downstream stages are below the weir elevation.

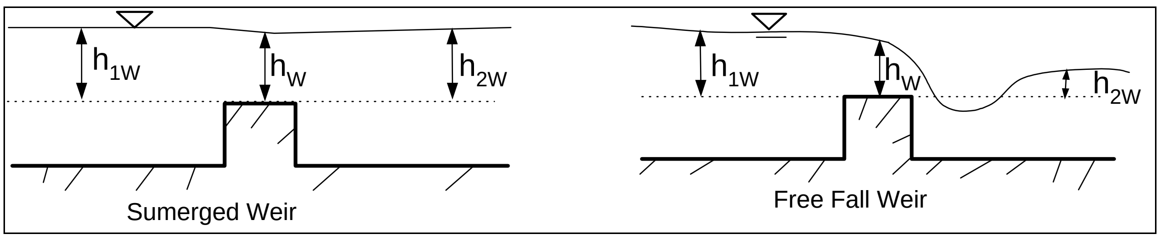

Fig. 3.1-9. Submerged versus Free Fall Weir.

Similarly, for two internal boundary nodes separated by a gate, $Q_{1G} = Q_{2G} = Q_G$. When the flow is not influenced by the gate opening (Fig. 3.1-10), the flow rate is given by

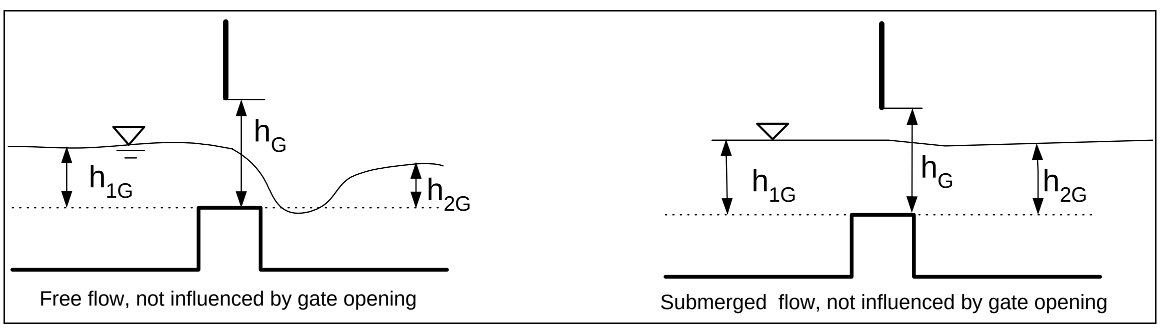

Fig. 3.1-10. Gate Opening Does Not Affect Flow.

$$Q_G = \frac{2}{3\sqrt{3}} C_G h_{1G} B_G \sqrt{h_{1G}} \quad if \quad h_{2G} < \frac{2}{3} h_{1G} \ and\ h_G > \frac{2}{3} h_{1G} \qquad \textbf{(3.1.81)}$$

$$Q_G = C_G B_G h_{2G} \sqrt{h_{1G} - h_{2G}} \quad if \quad h_{1G} > h_{2G} > \frac{2}{3} h_{1G} \ and\ h_G > \frac{2}{3} h_{1G} \qquad \textbf{(3.1.82)}$$

where C_G is the gate coefficient and B_G is the gate width [L]. When the gate opening affects the flow (Fig. 3.1-11), the flow rate is given by

$$Q_G = \frac{2}{3\sqrt{3}} C_G h_G B_G \sqrt{h_{1G}} \quad if \quad h_{2G} < \frac{2}{3} h_{1G} \ and\ h_G < \frac{2}{3} h_{1G} \qquad \textbf{(3.1.83)}$$

$$Q_G = C_G B_G h_G \sqrt{h_{1G} - h_{2G}} \quad if \quad h_{1G} > h_{2G} > \frac{2}{3} h_{1G} \; and \; h_G < \frac{2}{3} h_{1G} \qquad \textbf{(3.1.84)}$$

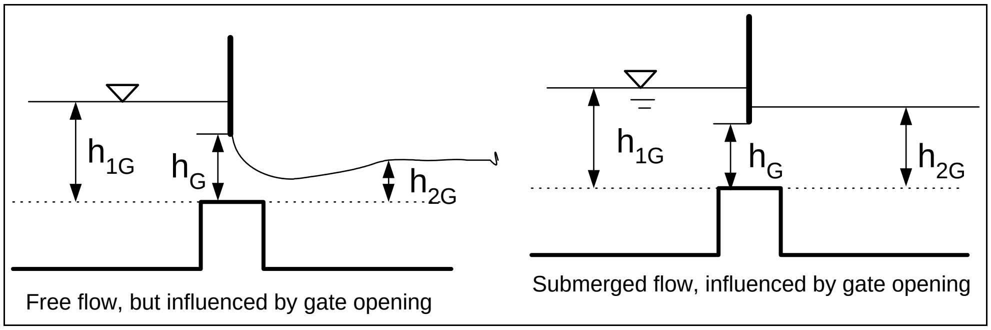

Fig. 3.1-11. Gate Opening Affects Flow.

For two internal boundary nodes separated by a culvert, $Q_{1C} = Q_{2C} = Q_C$. Various formulae for Q_C can be found in the literature.

3.1.2.2 The Hybrid Lagrangian-Eulerian Finite Element Method. When the hybrid Lagrangian-Eulerian finite element method is used to solve the diffusive wave equation, instead of Eq. (3.1.63), using the definition of Q = VA, we expand Eq. (2.1.1) to yield following diffusive wave equation in the Lagrangian form

$$\frac{D_V A}{D\tau} + KA = S_S + S_R - S_E + S_I + S_1 + S_2 \quad where \quad K = \frac{\partial V}{\partial x} \qquad \textbf{(3.1.85)}$$

To use the semi-Lagrangian method to solve the diffusive wave equation, we integrate Eq. (3.1.85) along its characteristic line from x_i at new time level to x_i^* at old time level or on the boundary (Fig. 3.1-12), we obtain

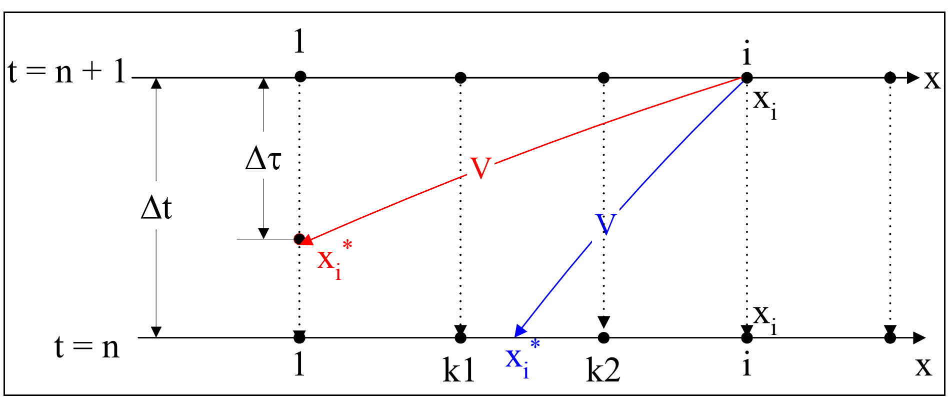

Fig. 3.1-12. Backward Particle Tracking in One Dimension.

$$\left(1+\frac{\Delta\tau}{2}K_i^{(n+1)}\right)A_i^{(n+1)}=\left(1-\frac{\Delta\tau}{2}K_i^{*}\right)A_i^{*}+\frac{\Delta\tau}{2}\left(S_{Si}^{(n+1)}+S_{Si}^{*}\right)+\frac{\Delta\tau}{2}\left(S_{Ri}^{(n+1)}+S_{Ri}^{*}\right)$$
$$-\frac{\Delta\tau}{2}\left(S_{Ei}^{(n+1)}+S_{Ei}^{*}\right)+\frac{\Delta\tau}{2}\left(S_{Ii}^{(n+1)}+S_{Ii}^{*}\right)+\frac{\Delta\tau}{2}\left(S_{1i}^{(n+1)}+S_{1i}^{*}\right)+\frac{\Delta\tau}{2}\left(S_{2i}^{(n+1)}+S_{2i}^{*}\right)$$

or analytically,

$$A_i^{(n+1)}=A_i^{*}e^{-\overline{K}\Delta\tau}+\frac{\overline{SS}}{\overline{K}}\left(1-e^{-\overline{K}\Delta\tau}\right)\quad or\quad A_i^{(n+1)}=\frac{\overline{SS}}{\overline{K}}+\left(A_i^{*}-\frac{\overline{SS}}{\overline{K}}\right)e^{-\overline{K}\Delta\tau} \qquad \textbf{(3.1.86)}$$

$$If\quad A_i^{(n+1)}<0\quad set\quad A_i^{(n+1}=0,\ \text{where}\quad \overline{K}=\frac{1}{2}\left(K_i^{(n+1)}+K_i^{*}\right)\quad and$$

$$\overline{SS}=\frac{1}{2}\left(\left(S_{Si}^{(n+1)}+S_{Ri}^{(n+1)}-S_{Ei}^{(n+1)}+S_{Ii}^{(n+1)}+S_{1i}^{(n+1)}+S_{2i}^{(n+1)}\right)+\left(S_{Si}^{*}+S_{Ri}^{*}-S_{Ei}^{*}+S_{Ii}^{*}+S_{1i}^{*}+S_{2i}^{*}\right)\right)$$

where $\Delta\tau$ is the tracking time, it is equal to Δt when the backward tracking is carried out all the way to the root of the characteristic and it is less than Δt when the backward tracking hits the boundary before Δt is consumed (Fig. 3.1-12); $K_i^{(n+1)}$, $A_i^{(n+1)}$, $S_{Si}^{(n+1)}$, $S_{Ri}^{(n+1)}$, $S_{Ei}^{(n+1)}$, $S_{Ii}^{(n+1)}$, $S_{1i}^{(n+1)}$, and $S_{2i}^{(n+1)}$ respectively, are the values of K, A, S_S, S_R, S_E, S_I, S_1, and S_2, respectively, at x_i at new time level $t = (n+1)\Delta t$; and K_i^*, A_i^*, S_{Si}^*, S_{Ri}^*, S_{Ei}^*, S_{Ii}^*, S_{1i}^*, and $\mathbf{S_{2i}}^*$, respectively, are the values of K, A, S_S, S_R, S_E, S_I, S_1, and S_2, respectively, at the location x_i^*. Since the velocity V and the decay coefficient K are functions of A, this is a nonlinear hyperbolic problem. Equation (3.1.86) is solved iteratively to yield the cross-sectional area A, and hence the water depth h. The iteration procedure is outlined as follows:

(i) Given the value of $A^{(k)}$ at the *k-th* iteration, compute h and H.

(ii) Apply finite element method to the following equation to obtain V

$$V=\frac{-a}{n}\left[\frac{R}{1+\left(\frac{\partial Z_0}{\partial x}\right)^2}\right]^{2/3}\frac{1}{\sqrt{\left|-\frac{\partial H}{\partial x}-\frac{h}{c\rho}\frac{\partial\Delta\rho}{\partial x}+\frac{B\tau^S}{Ag\rho}\right|}}\left(\frac{\partial H}{\partial x}+\frac{h}{c\rho}\frac{\partial\Delta\rho}{\partial x}-\frac{B\tau^S}{Ag\rho}\right) \qquad \textbf{(3.1.87)}$$

(iii) Perform particle tracking to locate x^* and obtain all the *-superscripted quantities.

(iv) Apply the finite element method to the following equation to obtain K

$$K=\frac{\partial V}{\partial x} \qquad \textbf{(3.1.88)}$$

(v) Solve Eq. (3.1.86) along with the boundary condition to obtain new $A^{(k+1)}$

(vi) Check if $A^{(k+1)}$ converges, if yes go to the next time step.

(vii) If $A^{(k+1)}$ does not converge, update A with $A^{(k)} \leftarrow \omega A^{(k+1)} + (1-\omega)A^{(k)}$ and repeat Steps (i) through (vi).

When the wave is transported out of the region at a boundary node (i.e., when $\mathbf{N}\bullet\mathbf{V} \geq 0$), a boundary condition is not needed. When the wave is transported into the region at a node (i.e., when $\mathbf{N}\bullet\mathbf{V} < 0$), a boundary condition must be specified. As in the Galerkin finite element method, three types of boundary conditions may be encountered.

Dirichlet boundary condition:

For the Dirichlet boundary, the water depth is prescribed, thus the cross sectional area, A, is computed from the relationship between the cross section area versus depth curve as

$$H_I = H_{Id}, \quad I \in N_D \Rightarrow A_I = A_{Id}, \quad I \in N_D \qquad \textbf{(3.1.89)}$$

Flux boundary condition:

For the flux boundary, the flow rate is prescribed as function of time at the boundary node, from which the boundary value is computed as

$$A^{(n+1)} = \frac{Q_{up}(t)}{V^{(n+1,k)}} \qquad \textbf{(3.1.90)}$$

where $Q_{up}(t)$, a function of time t, is the prescribed flow rate [L^3/t] and $V^{(n+1,k)}$ is the value of V at new time and previous iteration.

Water depth-dependent boundary condition: prescribed rating curve

For the boundary where a rating curve is used to describe the relationship between water depth, h, and the discharge, Q, the cross sectional area, A, on the boundary is computed with

$$V^{(n+1,k)} A^{(n+1)} = f(h) \qquad \textbf{(3.1.91)}$$

where $f(h)$ is the rating curve which is a function of h. Equation (3.1.91) is solved iteratively to yield $A^{(n+1)}$.

Junction Boundary Condition:

If the node IJ is an internal boundary node that connects a junction J, then H_{IJ} is a function of water depth, h_{IJ-1}, of its immediately internal node and of water surface at the junction J, H_J. This functional relationship is obtained by applying the finite element method to Eq. (3.1.63) to yield the governing equation for *Node IJ* similar to Eqs. (3.1.71) through (3.73)

$$C^1_{1J,1J-1} H^1_{1J-1} + C^1_{1J,1J} H^1_{1J} = L^1_{1J} + Q^1_{1J} + Q^1_{I1J} + Q^1_{11J} + Q^1_{21J} \qquad \textbf{(3.1.92)}$$

$$C^2_{2J,2J-1} H^2_{2J-1} + C^2_{2J,2J} H^2_{2J} = L^2_{2J} + Q^2_{2J} + Q^2_{I2J} + Q^2_{12J} + Q^2_{22J} \qquad \textbf{(3.1.93)}$$

$$C^3_{3J,3J-1} H^3_{3J-1} + C^3_{3J,3J} H^3_{3J} = L^3_{3J} + Q^3_{3J} + Q^3_{I3J} + Q^3_{13J} + Q^3_{23J} \qquad \textbf{(3.1.94)}$$

where the superscript denotes the reach number and subscript denotes node number in a reach, for example, H_{1J}^1 denotes the total head at the *1J-th* node in *Reach 1*. Equation (3.1.92) has two unknowns, H_{1J}^1 and Q_{1J}^1, the unknown H_{1J-1}^1 is obtained by inverting A_{1J-1}^1, which is obtained from particle tracking in *Reach 1*. Similarly, Equation (3.1.93) has two unknowns, H_{2J}^{2} and Q_{2J}^{2}, and Equation (3.1.94) has two unknowns, H_{3J}^{3} and Q_{3J}^{3}. The number of unknowns (6) is more than the number of equations (3). Three more governing equations must be set up, which can be obtained based on the continuity of energy lines. This is described as follows.

Assume the entrance loss to the junction and exit loss from the junction are negligible, we have the following three equations

$$H_{1j}^{1} + \frac{1}{2g}\left(\frac{Q_{1J}^{1}}{A_{1J}^{1}}\right)^{2} \quad = \quad h_J + Z_{oJ} \qquad \textbf{(3.1.95)}$$

$$H_{2j}^{2} + \frac{1}{2g}\left(\frac{Q_{2J}^{2}}{A_{2J}^{2}}\right)^{2} \quad = \quad h_J + Z_{oJ} \qquad \textbf{(3.1.96)}$$

$$H_{3j}^{3} + \frac{1}{2g}\left(\frac{Q_{3J}^{3}}{A_{3J}^{3}}\right)^{2} \quad = \quad h_J + Z_{oJ} \qquad \textbf{(3.1.97)}$$

where A_{1J}^{1}, A_{2J}^{2}, and A_{3J}^{3} are the cross-sectional area at Nodes *1J* of Reach *1*, Node *2J* of Reach *2*, and Node *3J* of Reach *3*, respectively; h_J is the water depth at the Junction *J*; and Z_{oJ} is the bottom elevation at the Junction *J*. It is noted that the second terms on the left hand side of Eqs. (3.1.95) through (3.1.97) are generally ignored in computation implementation to give more robust solutions.

The water depth at Junction *J* is not decoupled from river/stream/canal reaches. The water budget equation for the Junction J is

$$\frac{d\mathcal{V}_J}{dh_J}\frac{dh_J}{dt} = \sum_{i=1}^{i=3} Q_{iJ}^{i} \qquad \textbf{(3.1.98)}$$

When $\frac{d\mathcal{V}_J}{dh_J}$ is small, the water budget Eq. (3.1.98) is not employed. Instead, the following equation, resulting from the requirement that the summation of flow rates is equal to zero, is used

$$\sum_{i=3}^{i=3} Q_{iJ}^{i} = 0 \qquad \textbf{(3.1.99)}$$

Equations (3.1.92) through (3.1.97) and Eq. (3.1.98) or Eq. (3.1.99) constitute 7 equations for seven unknowns, A_{1J}^{1}, A_{2J}^{2}, A_{3J}^{3}, Q_{1J}^{1}, Q_{2J}^{2}, Q_{3J}^{3}, and h_J. These equations should be solved iteratively along with particle tracking for all internal nodes of the three reaches connecting the junction node J. The seven linearized equations can be solved with the Gaussian direct elimination with full pivoting.

Control structure boundary condition:

To facilitate the implementation of internal boundary conditions of control structures, we discretize the two internal boundary nodes of every structure with the finite element method. Then we can implement the boundary conditions similar to that in finite element modeling of diffusive wave approaches.

3.1.3 The Semi-Lagrangian Method for Kinematic Wave

To use the Lagrangian method to solve the kinematic wave equation, Eq. (2.1.65) is rewritten in the Lagrangian form as follows

$$\frac{D_V A}{D\tau} + KA = S_S + S_R - S_E + S_I + S_1 + S_2 \quad where \quad K = \frac{\partial V}{\partial x} \qquad \textbf{(3.1.100)}$$

in which K is the decay coefficient of the wave and S is the source/sink of the wave. Integrating Eq. (3.1.100) along its characteristic line from x_i at new time level to x_i^* (Fig. 3.1-12), we obtain

$$\left(1+\frac{\Delta\tau}{2}K_i\right)A_i^{(n+1)} = \left(1-\frac{\Delta\tau}{2}K_i^*\right)A_i^* + \frac{\Delta\tau}{2}\left(S_{Si}^{(n+1)}+S_{Si}^*\right) + \frac{\Delta\tau}{2}\left(S_{Ri}^{(n+1)}+S_{Ri}^*\right)$$
$$-\frac{\Delta\tau}{2}\left(S_{Ei}^{(n+1)}+S_{Ei}^*\right) + \frac{\Delta\tau}{2}\left(S_{Ii}^{(n+1)}+S_{Ii}^*\right) + \frac{\Delta\tau}{2}\left(S_{Ii}^{(n+1)}+S_{Ii}^*\right) + \frac{\Delta\tau}{2}\left(S_{2i}^{(n+1)}+S_{2i}^*\right)$$

or analytically,

$$A_i^{(n+1)} = A_i^* e^{-\overline{K}\Delta\tau} + \frac{\overline{SS}}{\overline{K}}\left(1-e^{-\overline{K}\Delta\tau}\right) \quad or \quad A_i^{(n+1)} = \frac{\overline{SS}}{\overline{K}} + \left(A_i^* - \frac{\overline{SS}}{\overline{K}}\right)e^{-\overline{K}\Delta\tau} \qquad \textbf{(3.1.101)}$$

$$If \quad A_i^{(n+1)} < 0 \quad set \quad A_i^{(n+1} = 0, \text{ where } \quad \overline{K} = \frac{1}{2}\left(K_i^{(n+1)} + K_i^*\right) \quad and$$

$$\overline{SS} = \frac{1}{2}\left(\left(S_{Si}^{(n+1)} + S_{Ri}^{(n+1)} - S_{Ei}^{(n+1)} + S_{Ii}^{(n+1)} + S_{1i}^{(n+1)} + S_{2i}^{(n+1)}\right) + \left(S_{Si}^* + S_{Ri}^* - S_{Ei}^* + S_{Ii}^* + S_{1i}^* + S_{2i}^*\right)\right)$$

where $\Delta\tau$ is the tracking time, it is equal to Δt when the backward tracking is carried out all the way to the root of the characteristic and it is less than Δt when the backward tracking hits the boundary before Δt is consumed (Fig. 3.1-12); $K_i^{(n+1)}$, $A_i^{(n+1)}$, $S_{Si}^{(n+1)}$, $S_{Ri}^{(n+1)}$, $S_{Ei}^{(n+1)}$, $S_{Ii}^{(n+1)}$, $S_{1i}^{(n+1)}$, and $S_{2i}^{(n+1)}$ respectively, are the values of K, A, S_S, S_R, S_E, S_I, S_1, and S_2, respectively, at x_i at new time level $t = (n+1)\Delta t$; and K_i^*, A_i^*, S_{Si}^*, S_{Ri}^*, S_{Ei}^*, S_{Ii}^*, S_{1i}^*, and S_{2i}^*, respectively, are the values of K, A, S_S, S_R, S_E, S_I, S_1, and S_2, respectively, at the location x_i^*. Because of density and wind effects, the velocity V and the decay coefficient K are functions of A, this is nonlinear problem. However, because the nonlinearity due to density and wind effects are normally very weak, Equation (3.1.101) is considered a linear hyperbolic problem with the nonlinear effects evaluated using the values of A at previous time. This equation is used to compute the cross-sectional area A, and hence the water depth h, at all nodes except for the upstream boundary node.

Because the wave is transported into the region at an upstream node, a boundary condition must be specified. The flow rate is normally given as a function of time at an upstream node, from which the boundary value is computed as

$$A_i^{(n+1)} = \frac{Q_{up}(t)}{V_i^{(n+1)}} \tag{3.1.102}$$

where $Q_{up}(t)$, a function of time t, is the prescribed flow rate [L^3/t].

3.1.4 Numerical Approximations of Thermal Transport

Two options are provided in this report to solve the thermal transport equation. One is the finite element method and the other is the particle tracking method.

3.1.4.1 Finite Element Method. Recall the thermal transport equation is governed by Eq. (2.1.67) which is rewritten in a slightly different form as

$$\begin{aligned} &\rho_W C_W A \frac{\partial T}{\partial t} + \frac{\partial(\rho_W C_W A)}{\partial t} T + \frac{\partial(\rho_W C_W Q T)}{\partial x} - \frac{\partial}{\partial x}\left(D^H A \frac{\partial T}{\partial x} \right) \\ &= S_h^a + S_h^r + S_h^n - S_h^b - S_h^e - S_h^s + S_h^i + S_h^{o1} + S_h^{o2} + S_h^c \end{aligned} \tag{3.1.103}$$

Applying the finite element method to Eq. (3.1.103), we obtain the following matrix equation

$$\begin{aligned} &[M]\frac{d\{T\}}{dt} + [V]\{T\} + [D]\{T\} + [K]\{T\} = -\{\Phi_B\} + \{\Phi^a\} \\ &+ \{\Phi^r\} + \{\Phi^n\} - \{\Phi^b\} - \{\Phi^e\} - \{\Phi^s\} + \{\Phi^i\} + \{\Phi^{o1}\} + \{\Phi^{o2}\} + \{\Phi^c\} \end{aligned} \tag{3.1.104}$$

in which

$$\begin{aligned} &M_{ij} = \int_{X_1}^{X_N} N_i \rho_W C_W A N_j dx, \quad V_{ij} = \int_{X_1}^{X_N} \frac{dW_i}{dx} \rho_W C_W Q N_j dx, \quad D_{ij} = \int_{X_1}^{X_N} \frac{dN_i}{dx} D^H A \frac{dN_j}{dx} dx, \\ &K_{ij} = \int_{X_1}^{X_N} N_i \frac{\partial \rho_W C_W A}{\partial t} N_j dx, \quad \Phi_{Bi} = \left(W_i \rho_W C_W Q T - N_i D^H A \frac{\partial T}{\partial x} \right) \Bigg|_{x+X_1}^{x=X_N} \end{aligned} \tag{3.1.105}$$

$$\Phi_i^a = \int_{X_1}^{X_N} N_i S_h^a dx, \quad \Phi_i^r = \int_{X_1}^{X_N} N_i S_h^r dx, \quad \Phi_i^n = \int_{X_1}^{X_N} N_i S_h^n dx \tag{3.1.106}$$

$$\Phi_i^b = \int_{X_1}^{X_N} N_i S_h^b dx, \quad \Phi_i^e = \int_{X_1}^{X_N} N_i S_h^e dx, \quad \Phi_i^s = \int_{X_1}^{X_N} N_i S_h^s dx, \quad \Phi_i^c = \int_{X_1}^{X_N} N_i S_h^c dx, \tag{3.1.107}$$

$$\Phi_i^i = \int_{X_1}^{X_N} N_i S_h^i dx, \quad \Phi_i^{o1} = \int_{X_1}^{X_N} N_i S_h^{o1} dx, \quad \Phi_i^{o2} = \int_{X_1}^{X_N} N_i S_h^{o2} dx \tag{3.1.108}$$

where $W_i(x)$ is the weighting function of node at x_i; $N_i(x)$ and $N_j(x)$, functions of x, are the base functions of nodes at x_i and x_j, respectively; [M] is the mass matrix, [V] is the stiff matrix due to advective transport; [D] is the stiff matrix due to dispersion/diffusion/conduction; {T} is the solution vector of temperature; $\{\Phi_B\}$ is the vector due to boundary conditions, which can contribute to load vector and/or coefficient matrix; $\{\Phi^a\}$ is the load vector due to artificial energy source; $\{\Phi^r\}$ is the load vector due to energy in rainfall; $\{\Phi^n\}$ is the load vector due to net radiation; $\{\Phi^b\}$ is the vector due to backward radiation, which is a nonlinear function of temperature and contributes to both the load vector and coefficient matrix; $\{\Phi^e\}$ is the vector due to energy consumed for evaporation, which is a nonlinear function of temperature and contributes to both the load vector and coefficient matrix; $\{\Phi^s\}$ is the vector due to sensible heat, which is a linear function of temperature and contributes to both the load vector and coefficient matrix; $\{\Phi^c\}$ is the vector due to chemical reaction, which is not considered in this version, but can be added easily; $\{\Phi^i\}$ is the vector due to interaction with subsurface exfiltraing water; $\{\Phi^{o1}\}$ is the vector due to interaction with overland water via river bank 1; and $\{\Phi^{o2}\}$ is the vector due to interaction with overland water via river bank 2.

Approximating the time derivative term in Eq. (3.1.104) with a time-weighted finite difference, we reduce the advective-diffusive equation and its boundary conditions to the following matrix equation

$$[C]\{T\} = \{L\} - \{\Phi_B\} - \{\Phi^b\} - \{\Phi^e\} - \{\Phi^s\} + \{\Phi^i\} + \{\Phi^{o1}\} + \{\Phi^{o2}\} \quad \textbf{(3.1.109)}$$

in which

$$[C] = \frac{[M]}{\Delta t} + \theta([D] + [K]) + \theta_V [V],$$
$$\{L\} = \left(\frac{[M]}{\Delta t} - (1-\theta)([DS] + [K]) - (1-\theta_V)[V]\right)\{T^{(n)}\} + \{\Phi^a\} + \{\Phi^r\} + \{\Phi^n\} \quad \textbf{(3.1.110)}$$

where [C] is the coefficient matrix, {L} is the load vector from initial condition, artificial sink/sources, rainfall, and net radiation; Δt is the time step size; θ is the time weighting factor for the dispersion and linear terms; θ_v is the time weighting factor for the velocity term; and $\{T^{(n)}\}$ is the value of {T} at old time level n. The global and internal boundary (junctions, weirs, and gates) conditions must be used to provide $\{\Phi_B\}$ in Eq. (3.1.109). The interaction between the overland and river/stream/canal flows must be implemented to evaluate $\{\Phi^{o1}\}$ and $\{\Phi^{o2}\}$; and the interaction between the subsurface and river/stream/canal flows must be implemented to calculate $\{\Phi^i\}$. The interactions will be addressed in Section 3.4.

For a global boundary node I, the corresponding algebraic equation from Eq. (3.1.109) is

$$C_{I,I-1}T_{I-1} + C_{I,I}T_I + C_{I,I+1}T_{I+1} = L_I - \left(\Phi_I^b + \Phi_I^e + \Phi_I^s\right) + \left(\Phi_I^i + \Phi_I^{o1} + \Phi_I^{o2}\right) - \Phi_{BI} \quad \textbf{(3.1.111)}$$

In the above equations there are two unknowns T_I and Φ_{BI}; either T_I or Φ_{BI}, or the relationship between T_I and Φ_{BI} must be specified. The numerical implementation of these boundary conditions is described as follows.

Direchlet boundary condition: prescribed temperature

If T_I is given on the boundary node I (Dirichlet boundary condition), all coefficients ($C_{I,I-1}$, $C_{I,I}$, $C_{I,I+1}$) and right-hand side (L_I, Φ_I^b, Φ_I^e, Φ_I^s, Φ_I^i, Φ_I^{o1}, Φ_I^{o2}) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then an identity equation is created as

$$T_1 = T_{Id}, \quad I \in N_D \qquad \textbf{(3.1.112)}$$

where T_{Id} is the prescribed temperature on the Dirichlet node I and N_D is the number of Dirichlet boundary nodes. This process is repeated for every Dirichlet nodes. Note it is unnecessary to modify other equations that involving these unknowns, which was done in the previous version. By not modifying other equations, the symmetrical property of the matrix is preserved, which makes the iterative solvers more robust. The final set of equations will consist of N_D identity equations and (N - N_D) finite element equations for N unknowns T_i's. After T_i's for all nodes are solved from the matrix equation, Eq. (3.1.111) is then used to back calculate N_D Φ_{BI}'s.

If a direct solver is used to solve the matrix equation, the above procedure will solve N T_i's accurately except for roundoff errors. However, if an iterative solver is used, stopping criteria must be strict enough so that the converged solutions of N T_i's are accurate enough to the exact solution. With such accurate T_i's, then can be sure that the back-calculated ND Φ_{BI}'s are accurate.

Cauchy boundary condition: prescribed heat flux

If Φ_{BI} is given (Cauchy flux boundary condition), all coefficients ($C_{I,I-1}$, $C_{I,I}$, $C_{I,I+1}$) and right-hand side (L_I, Φ_I^a, Φ_I^r, Φ_I^n, Φ_I^i, Φ_I^{o1}, Φ_I^{o2}) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then Eq. (3.1.111) is modified to incorporate the boundary conditions and used to solve for T_I. The modification of Eq. (3.1.111) is straightforward. Because Φ_{BI} is a known quantity, it contributes to the load on the right hand side. This type of boundary conditions is very easy to implement. After T_i's are obtained, the original Eq. (3.1.111), which is stored in a temporary array, is used to back calculate N_C Φ_{BI}'s on flux boundaries (where N_C is the number of flux boundary nodes). These back-calculated Φ_{BI}'s should be theoretically identical to the input Φ_{BI}'s. However, because of round-off errors (in the case of direct solvers) or because of stopping criteria (in the case of iterative solvers), the back-calculated Φ_{BI}'s will be slightly different from the input Φ_{BI}'s. If the differences between the two are significant, it is an indication that the solvers have not yielded accurate solutions.

Neumann boundary condition: prescribed gradient of temperature

At Neumann boundaries, the temperature gradient is prescribed, thus, the flux due to temperature gradient is given. For this case, all coefficients ($C_{I,I-1}$, $C_{I,I}$, $C_{I,I+1}$) and right-hand side (L_I, Φ_I^a, Φ_I^r, Φ_I^n, Φ_I^i, Φ_I^{o1}, Φ_I^{o2}) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then Eq. (3.1.111) is modified to incorporate the boundary conditions and used to solve for T_I. For the Neumann boundary condition, Φ_{BI} contributes to both the matrix coefficient and load vector, thus both the coefficient matrix [C] and the load vector {L} must be modified. Recall

$$\Phi_{Bi} = \left(W_i \rho_W C_W Q T - N_i D^H A \frac{\partial T}{\partial x} \right) \Bigg|_{x=X_1}^{x=X_N} \qquad \textbf{(3.1.113)}$$

Apply this equation to Node I, we have

$$\Phi_{BI} \equiv n_I \rho_W C_W Q T_I - n_I D^H A \frac{\partial T}{\partial x}\Big|_{x=X_I} = n_I \rho_W C_W Q T_I - \Phi_{nbI} \tag{3.1.114}$$

where n_I is the unit outward normal vector at the boundary node I, Φ_{nbI} is the Neumann boundary flux at node I. Substitution of Eq. (3.1.114) into Eq. (3.1.111), we have the modified coefficient matrix and load vector; thus the modified Eq. (3.1.111). This modified equation is used to solve T_I. After T_I is solved, the original Eq. (3.1.111) (recall the original Eq. (3.1.111) must be and has been stored in a temporary array) is used to back-calculate Φ_{BI}.

Variable Boundary Condition:

At the variable boundary condition Node I, the implementation of boundary conditions can be made identical to that for a Cauchy boundary condition node if the flow is directed into the river/stream/canal reach. If the flow is going out of the reach, the boundary condition is implemented similar to the implementation of Neuman boundary condition with $\Phi_{nbI} = 0$. The assumption of zero Neumann flux implies that a Neuman node must be far away from the source/sink.

Junction boundary condition:

If the node IJ is an internal node that connects a junction J, then node IJ is treated as an internal boundary node. For example, consider three reaches with three internal nodes connecting to the junction J (Fig. 3.1-8). After applying the finite element method to Eq. (3.1.103), we have a total of (1J + 2J + 3J) algebraic equations. The algebraic equations for Nodes 1J, 2J, and 3J can be written based on Eq. (3.1.111)

$$C^1_{1J,1J-1} T^1_{1J-1} + C^1_{1J,1J} T^1_{1J} = L^1_{1J} - \left(\Phi^{b1}_{1J} + \Phi^{e1}_{1J} + \Phi^{s1}_{1J}\right) + \left(\Phi^{i1}_{1J} + \Phi^{o11}_{1J} + \Phi^{o21}_{1J}\right) - \Phi^1_{1J} \quad \textbf{(3.1.115}$$

$$C^2_{2J,2J-1} T^2_{2J-1} + C^2_{2J,2J} T^2_{2J} = L^2_{2J} - \left(\Phi^{b2}_{2J} + \Phi^{e2}_{2J} + \Phi^{s2}_{2J}\right) + \left(\Phi^{i2}_{2J} + \Phi^{o12}_{2J} + \Phi^{o22}_{2J}\right) - \Phi^2_{2J} \quad \textbf{(3.1.116}$$

$$C^3_{3J,3J-1} T^3_{3J-1} + C^3_{3J,3J} T^3_{3J} = L^3_{3J} - \left(\Phi^{a3}_{3J} + \Phi^{r3}_{3J} + \Phi^{n3}_{3J}\right) + \left(\Phi^{i3}_{3J} + \Phi^{o13}_{3J} + \Phi^{o23}_{3J}\right) - \Phi^3_{3J} \quad \textbf{(3.1.117}$$

where the superscript denotes the reach number and subscript denotes local node number in a reach, for example, T_{1J}^1 denotes the temperature at the 1J-th node in Reach 1. For a convenient discussion, let us associate each of the unknowns, T_1^1, - - T_{1J-1}^1 to each of the 1J-1 finite element equations in Reach 1. Similarly, we associate each of the unknowns, T_1^2, - - T_{2J-2}^2 to each of the 2J-1 finite element equations in Reach 2 and each of the unknowns and T_1^3, - - T_{3J-1}^3 to each of the 3J-1 finite element equations in Reach 3. The unknown, Φ_{1J}^1, Φ_{2J}^2, and Φ_{3J}^3, are absent from these (1J-1 + 2J-1 + 3J-1) equations. In other words, we can say each equation governs one unknown. However, two unknowns, T_{1J}^1 and Φ_{1J}^1, appear in Eq. (3.1.115). Similarly, Equation (3.1.116) has two unknowns, T_{2J}^2 and Φ_{2J}^2, and Equation (3.1.117) has two unknowns, T_{3J}^3 and Φ_{3J}^3. The number of unknowns, (1J + 2J + 3J) temperatures and Φ_{1J}^1, Φ_{2J}^2, and Φ_{3J}^3, is more than the number of equations, (1J + 2J + 3J) finite element equations. Three more governing equations must be set up, which can be obtained with the assumption that the energy flux is due mainly to advection as

$$\Phi_{1J}^{1} \equiv \left(\rho_W C_W Q T - D^H A \frac{\partial T}{\partial x} \right) \Big|_{1J}$$
$$= \rho_W C_W \frac{1}{2} Q_{1J}^{1} \left[\left(1 + sign\left(Q_{1J}^{1}\right)\right) T_{1J}^{1} + \left(1 - sign\left(Q_{1J}^{1}\right)\right) T_J \right] \quad \textbf{(3.1.118)}$$

$$\Phi_{2J}^{2} \equiv \left(\rho_W C_W Q T - D^H A \frac{\partial T}{\partial x} \right) \Big|_{2J}$$
$$= \rho_W C_W \frac{1}{2} Q_{2J}^{2} \left[\left(1 + sign\left(Q_{2J}^{2}\right)\right) T_{2J}^{2} + \left(1 - sign\left(Q_{2J}^{2}\right)\right) T_J \right] \quad \textbf{(3.1.119)}$$

$$\Phi_{3J}^{3} \equiv \left(\rho_W C_W Q T - D^H A \frac{\partial T}{\partial x} \right) \Big|_{3J}$$
$$= \rho_W C_W \frac{1}{2} Q_{3J}^{3} \left[\left(1 + sign\left(Q_{3J}^{3}\right)\right) T_{3J}^{3} + \left(1 - sign\left(Q_{3J}^{3}\right)\right) T_J \right] \quad \textbf{(3.1.120}$$

where Q_{1J}^{1}, Q_{2J}^{2}, and Q_{3J}^{3}, respectively, are the volumetric flow rates from/to Nodes 1J, 2J, and 3J, respectively, to/from the junction J [cf. Eqs. (3.1.71), (3.1.72), and (3.1.73), respectively].

Equations (3.1.118) through (3.1.120) introduce one additional unknown, T_J. One additional equation must be set up which can be done based on the energy budget at the junction J. The rate of change of energy at the junction J must be equal to the net energy rate from all reaches that join at J. This energy budget can be written as

$$\frac{d\left(\rho_W C_W V_J T_J\right)}{dt} = \sum_i \Phi_{iJ}^{i} \quad \textbf{(3.1.121)}$$

When the storage effect of the junction is small, the energy budget Eq. (3.1.121) is not employed. Instead, the following equation, resulting from the requirement that the summation of heat flux is equal to zero, is used

$$\sum_{i=1}^{i=3} \Phi_{iJ}^{i} = 0 \quad \textbf{(3.1.122)}$$

Equations (3.1.115) through (3.1.120) and Eq. (3.1.121) or Eq. (3.1.122) constitute 7 equations for seven unknowns, T_{1J}^{1}, T_{2J}^{2}, T_{3J}^{3}, Φ_{1J}^{1}, Φ_{2J}^{2}, Φ_{3J}^{3}, and T_J. If there are N_J junctions, there will be N_J blocks of seven equations. These N_J blocks of equations should be solved iteratively along with N_R block of finite element equations where N_R is the number of reaches. In other words, the whole system of algebraic equations can be solved with block iterations. Each block of equations can be solved directly. For example, each of N_R blocks of finite element equations can be solved with an efficient tri-diagonal matrix solver such as the Thomas algorithm. Each of the N_J blocks of seven equations can be solved with the Gaussian direct elimination with full pivoting.

Control structure boundary condition:

The control structures may include weirs, gates, culverts, etc. For the two internal boundary nodes 1S and 2S separated by a structure, the boundary conditions at these two nodes are given by

$$\Phi_{1S} \equiv \left(\rho_W C_W Q T - D^H A \frac{\partial T}{\partial x} \right) \bigg|_{1S} = \rho_W C_W \frac{1}{2} Q \left[(1 + sign(Q)) T_{1S} + (1 - sign(Q)) T_{2S} \right] \quad \textbf{(3.1.123)}$$

$$\Phi_{2S} \equiv \left(\rho_W C_W Q T - D^H A \frac{\partial T}{\partial x} \right) \bigg|_{2S} = \rho_W C_W \frac{1}{2} Q \left[(1 + sign(Q)) T_{1S} + (1 - sign(Q)) T_{2S} \right] \quad \textbf{(3.1.124)}$$

where Φ_{1S} is the energy flux through node 1S; Φ_{2s} is the energy flux through node 2S; and Q is the flow rate through the structure S; sign(Q) is equal 1.0 if the flow is from node 1S to node 2S, -1.0 if flow is from node 2S to node 1S; T_{1S} is the temperature at node 1S; and T_{2S} is the temperature at node 2S.

3.1.4.2 The Hybrid Lagrangian-Eulerian Finite Element Method. When the hybrid Lagrangian-Eulerian finite element method is used to solve the thermal transport equation, we expand Eq. (3.1.103) to yield following advection-dispersion equation in the Lagrangian form

$$\frac{D_V T}{Dt} + KT = D + \Phi^S + \Phi^I + \Phi^{O1} + \Phi^{O2} \quad where \quad V = \frac{Q}{A} \quad \textbf{(3.1.125)}$$

in which

$$K = \frac{1}{\rho_W C_W A} \frac{\partial \rho_W C_W A}{\partial t} + \frac{1}{\rho_W C_W A} \frac{\partial \rho_W C_W Q}{\partial x}, \quad D = \frac{1}{\rho_W C_W A} \frac{\partial}{\partial x} \left(D^H A \frac{\partial T}{\partial x} \right)$$

$$\Phi^S = \frac{S_h^a + S_h^r + S_h^n - S_h^b - S_h^e - S_h^s}{\rho_W C_W A}, \quad \textbf{(3.1.126)}$$

$$and \quad \Phi^I = \frac{S_h^i}{\rho_W C_W A}, \quad \Phi^{O1} = \frac{S_h^{o1}}{\rho_W C_W A}, \quad \Phi^{O2} = \frac{S_h^{o2}}{\rho_W C_W A}$$

To use the semi-Lagrangian method to solve the thermal transport equation, we integrate Eq. (3.1.125) along its characteristic line from x_i at new time level to x_i^* (Fig. 3.1-12), we obtain

$$\begin{aligned} \left(1 + \frac{\Delta\tau}{2} K_i^{(n+1)} \right) T_i^{(n+1)} &= \left(1 - \frac{\Delta\tau}{2} K_i^* \right) T_i^* + \frac{\Delta\tau}{2} \left(D_i^{(n+1)} + D_i^* \right) + \frac{\Delta\tau}{2} \left(\Phi_i^{S^{(n+1)}} + \Phi_i^{S^*} \right) \\ &+ \frac{\Delta\tau}{2} \left(\Phi_i^{I^{(n+1)}} + \Phi_i^{I^*} \right) + \frac{\Delta\tau}{2} \left(\Phi_i^{O1^{(n+1)}} + \Phi_i^{O1^*} \right) + \frac{\Delta\tau}{2} \left(\Phi_i^{O2^{(n+1)}} + \Phi_i^{O2^*} \right), \quad i \in N \end{aligned} \quad \textbf{(3.1.127)}$$

where $\Delta\tau$ is the tracking time, it is equal to Δt when the backward tracking is carried out all the way to the root of the characteristic and it is less than Δt when the backward tracking hits the boundary before Δt is consumed; $K_i^{(n+1)}$, $T_i^{(n+1)}$, $D_i^{(n+1)}$, $\Phi_i^{S(n+1)}$, $\Phi_i^{I(n+1)}$, $\Phi_i^{O1(n+1)}$, and $\Phi_i^{O2(n+1)}$ respectively, are the values of K, T, D, Φ^S, Φ^I, Φ^{O1}, and Φ^{O2}, respectively, at x_i at new time level $t = (n+1)\Delta t$; and K_i^*, T_i^*, D_i^*, Φ_i^{S*}, Φ_i^{I*}, Φ_i^{O1*}, and Φ_i^{O2*}, respectively, are the values of K, T, D, Φ^S, Φ^I, Φ^{O1}, and Φ^{O2}, respectively, at the location x_i^*.

To compute the dispersion/diffusion terms $D_i^{(n+1)}$ and D_i^*, we rewrite the second equation in Eq.

(3.1.126) as

$$\rho_W C_W AD = \frac{\partial}{\partial x}\left(D^H A \frac{\partial T}{\partial x} \right) \qquad \textbf{(3.1.128)}$$

Applying the Galerkin finite element method to Eq. (3.1.128) at new time level (n+1), we obtain the following matrix equation for $\{D^{(n+1)}\}$ as

$$\left[a^{(n+1)}\right]\left\{D^{(n+1)}\right\} + \left[b^{(n+1)}\right]\left\{T^{(n+1)}\right\} = \left\{B^{(n+1)}\right\} \qquad \textbf{(3.1.129)}$$

in which

$$\left\{D^{(n+1)}\right\} = \left\{D_1^{(n+1)} \quad D_2^{(n+1)} \quad . \; . \; D_i^{(n+1)} \quad . \; . \; D_N^{(n+1)}\right\}^{Transpose} \qquad \textbf{(3.1.130)}$$

$$\left\{T^{(n+1)}\right\} = \left\{T_1^{(n+1)} \quad T_2^{(n+1)} \quad . \; . \; T_i^{(n+1)} \quad . \; . \; T_N^{(n+1)}\right\}^{Transpose} \qquad \textbf{(3.1.131)}$$

$$\left\{B^{(n+1)}\right\} = \left\{B_1^{(n+1)} \quad B_2^{(n+1)} \quad . \; . \; B_i^{(n+1)} \quad . \; . \; B_N^{(n+1)}\right\}^{Transpose} \qquad \textbf{(3.1.132)}$$

$$a_{ij}^{(n+1)} = \int_{X_1}^{X_N} N_i \left(\rho_W C_W A\right)\Big|_{(n+1)} N_j dx, \quad b_{ij}^{(n+1)} = \int_{X_1}^{X_N} \frac{dN_i}{dx}\left(D^H A\right)\Big|_{(n+1)} \frac{dN_j}{dx} dx,$$

$$B_i^{(n+1)} = N_i \left(D^H A\right)\Big|_{(n+1)} \frac{\partial T^{(n+1)}}{\partial x}\Big|_{X=X_1}^{X=X_N} \qquad \textbf{(3.1.133)}$$

where the superscript (n+1) denotes the time level; N_i and N_j are the base functions of nodes at x_i and x_j, respectively.

Lumping the matrix $[a^{(n+1)}]$, we can solve Eq. (3.1.129) for $D_I^{(n+1)}$ as follows

$$D_I^{(n+1)} = -\frac{1}{a_{II}^{(n+1)}} \sum_j b_{Ij}^{(n+1)} T_j^{(n+1)} \quad if \quad I \in \{2,3,..,N-1\}$$

$$D_I^{(n+1)} = \frac{1}{a_{II}^{(n+1)}} B_1^{(n+1)} - \frac{1}{a_{II}^{(n+1)}} \sum_j b_{Ij}^{(n+1)} T_j^{(n+1)} \quad if \quad I \in \{1,N\} \qquad \textbf{(3.1.134)}$$

where $a_{II}^{(n+1)}$ is the lumped $a_{ii}^{(n+1)}$. Following the identical procedure that leads Eq. (3.1.128) to Eq. (3.1.134), we have

$$D_I^{(n)} = -\frac{1}{a_{II}^{(n)}} \sum_j b_{Ij}^{(n)} T_j^{(n)} \quad if \quad I \in \{2,3,..,N-1\}$$

$$D_I^{(n)} = \frac{1}{a_{II}^{(n)}} B_1^{(n)} - \frac{1}{a_{II}^{(n)}} \sum_j b_{Ij}^{(n)} T_j^{(n)} \quad if \quad I \in \{1,N\} \qquad \textbf{(3.1.135)}$$

where $\{B^{(n)}\}$, $\{a^{(n)}\}$ and $\{b^{(n)}\}$, respectively, are defined similar to $\{B^{(n+1)}\}$, $\{a^{(n+1)}\}$ and $\{b^{(n+1)}\}$, respectively.

With $\{D^{(n)}\}$ calculated with Eq. (3.1.135), $\{D^*\}$ can be interpolated. Substituting Eq. (3.1.134) into

Eq. (3.1.127) and implementing boundary conditions given in Section 2.1.4, we obtain a system of N simultaneous algebraic equations N unknowns ($T_i^{(n+1)}$ for i = 1, 2, .., N.) If the dispersion/diffusion term is not included, then Eq. (3.1.127) is reduced to a set of N decoupled equations as

$$a_{ii} T_i^{(n+1)} = b_i, \; i \in N \tag{3.1.136}$$

where

$$a_{ii} = \left(1 + \frac{\Delta \tau}{2} K_i^{(n+1)} \right) \tag{3.1.137}$$

$$\begin{aligned} b_i = & \left(1 - \frac{\Delta \tau}{2} K_i^* \right) T_i^* + \frac{\Delta \tau}{2} \left(\Phi_i^{S^{(n+1)}} + \Phi_i^{S*} \right) \\ & + \frac{\Delta \tau}{2} \left(\Phi_i^{I^{(n+1)}} + \Phi_i^{I*} \right) + \frac{\Delta \tau}{2} \left(\Phi_i^{O1^{(n+1)}} + \Phi_i^{O1*} \right) + \frac{\Delta \tau}{2} \left(\Phi_i^{O2^{(n+1)}} + \Phi_i^{O2*} \right), \quad i \in N \end{aligned} \tag{3.1.138}$$

Equations (3.1.136) is applied to all interior nodes without having to make any modification. On a boundary point, there two possibilities: Eq. (3.1.136) is replaced with a boundary equations when the flow is directed into the reach or Eq. (3.1.136) is still valid when the flow is direct out of the reach. In other words, when the thermal energy is transported out of the region at a boundary node (i.e., when $\mathbf{n} \bullet \mathbf{V} \geq 0$), a boundary condition is not needed and Equation (3.1.136) is used to compute the $T_i^{(n+1)}$. When the thermal energy is transported into the region at a node (i.e., when $\mathbf{n} \bullet \mathbf{V} < 0$), a boundary condition must be specified.

Alternatively, to facilitate the implementation of boundary condition at incoming flow node, the algebraic equation for the boundary node is obtained by applying the finite element method to the boundary node. For this alternative approach, the implementation of boundary conditions at global boundary nodes, internal junction nodes, and internal nodes connecting to control structures is identical to that in the finite element approximation of solving the thermal transport equation.

3.1.5 Numerical Approximations of Salinity Transport

Two options are provided in this report to solve the salinity transport equation. One is the finite element method and the other is the particle tracking method.

3.1.5.1 Finite Element Method. Recall the salinity transport equation is governed by Eq. (2.1.86) which is rewritten in a slightly different form as

$$A \frac{\partial S}{\partial t} + \frac{\partial A}{\partial t} S + \frac{\partial (QS)}{\partial x} - \frac{\partial}{\partial x} \left(D^S A \frac{\partial S}{\partial x} \right) = M_S^a + M_S^r + M_S^i + M_S^{o1} + M_S^{o2} \tag{3.1.139}$$

Applying the finite element method to Eq. (3.1.139), we obtain the following matrix equation

$$\begin{aligned} & [M] \frac{d\{S\}}{dt} + [V]\{S\} + [D]\{S\} + [K]\{S\} \\ & \quad = -\{\Psi^B\} + \{\Psi^a\} + \{\Psi^T\} + \{\Psi^i\} + \{\Psi^{o1}\} + \{\Psi^{o2}\} \end{aligned} \tag{3.1.140}$$

in which

$$M_{ij} = \int_{X_1}^{X_N} N_i A N_j dx, \quad V_{ij} = \int_{X_1}^{X_N} \frac{dW_i}{dx} Q N_j dx, \quad D_{ij} = \int_{X_1}^{X_N} \frac{dN_i}{dx} D^S A \frac{dN_j}{dx} dx,$$

$$K_{ij} = \int_{X_1}^{X_N} N_i \frac{\partial A}{\partial t} N_j dx, \quad \Psi_i^B = \left(W_i QS - N_i D^S A \frac{\partial T}{\partial x} \right) \Big|_{x=X_1}^{x=X_N} \tag{3.1.141}$$

$$\Psi_i^a = \int_{X_1}^{X_N} N_i M_s^a dx, \quad \Psi_i^r = \int_{X_1}^{X_N} N_i M_s^r dx \tag{3.1.142}$$

$$\Psi_i^i = \int_{X_1}^{X_N} N_i M_s^i dx, \quad \Psi_i^{o1} = \int_{X_1}^{X_N} N_i M_s^{o1} dx, \quad \Psi_i^{o2} = \int_{X_1}^{X_N} N_i M_s^{o2} dx \tag{3.1.143}$$

where W_i is the weighting function of node at x_i; N_i and N_j are the base functions of nodes at x_i and x_j, respectively; [M] is the mass matrix, [V] is the stiff matrix due to advective transport; [D] is the stiff matrix due to dispersion/diffusion/conduction; [K] is the stiff matrix due to the linear term; {S} is the solution vector of salinity; $\{\Psi^B\}$ is the vector due to boundary conditions, which can contribute to load vector and/or coefficient matrix; $\{\Psi^a\}$ is the load vector due to artificial salt source; $\{\Psi^r\}$ is the load vector due to salt in rainfall; $\{\Psi^i\}$ is the vector due to interaction with subsurface exfiltraing water; $\{\Psi^{o1}\}$ is the vector due to interaction with overland water via river bank 1; and $\{\Psi^{o2}\}$ is the vector due to interaction with overland water via river bank 2.

Approximating the time derivative term in Eq. (3.1.140) with a time-weighted finite difference, we reduce the advective-diffusive equation and its boundary conditions to the following matrix equation.

$$[C]\{S\} = \{L\} - \{\Psi^B\} - \{\Psi^i\} + \{\Psi^{o1}\} + \{\Psi^{o2}\} \tag{3.1.144}$$

in which

$$[C] = \frac{[M]}{\Delta t} + \theta([D] + [K]) + \theta_V [V],$$

$$\{L\} = \left(\frac{[M]}{\Delta t} - (1-\theta)([D] + [K]) - (1-\theta_V)[V] \right) \{S^{(n)}\} + \{\Psi^a\} + \{\Psi^r\} \tag{3.1.145}$$

where [C] is the coefficient matrix, {L} is the load vector from initial condition, artificial sink/sources and rainfall; Δt is the time step size; θ is the time weighting factor for the dispersion and linear terms; θ_v is the time weighting factor for the velocity term; and $\{S^{(n)}\}$ is the value of {S} at old time level n. The global and internal boundary (junctions, weirs, and gates) conditions must be used to provide $\{\Phi_B\}$ in Eq. (3.1.144). The interaction between the overland and river/stream/canal flows must be implemented to evaluate $\{\Psi^{o1}\}$ and $\{\Psi^{o2}\}$; and the interaction between the subsurface and river/stream/canal flows must be implemented to calculate $\{\Psi^i\}$. The interactions will be addressed in Section 3.4.

For a global boundary node I, the corresponding algebraic equation from Eq. (3.1.144) is

$$C_{I,I-1}S_{I-1} + C_{I,I}S_I + C_{I,I+1}S_{I+1} = L_I + \left(\Psi_I^i + \Psi_I^{o1} + \Psi_I^{o2}\right) - \Psi_I^B \quad \textbf{(3.1.146)}$$

In the above equations there are two unknowns T_I and Φ_{BI}; either T_I or Φ_{BI}, or the relationship between T_I and Ψ_I^B must be specified. The numerical implementation of these boundary conditions is described as follows.

Direchlet boundary condition: prescribed salinity

If S_I is given on the boundary node I (Dirichlet boundary condition), all coefficients ($C_{I,I-1}$, $C_{I,I}$, $C_{I,I+1}$) and right-hand side (L_I, Ψ_I^i, Ψ_I^{o1}, Ψ_I^{o2}) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then an identity equation is created as

$$S_I = S_{Id}, \quad I \in N_D \quad \textbf{(3.1.147)}$$

where S_{Id} is the prescribed salinity on the Dirichlet node I and N_D is the number of Dirichlet boundary nodes. This process is repeated for every Dirichlet nodes. Note it is unnecessary to modify other equations that involving these unknowns, which was done in the previous version. By not modifying other equations, the symmetrical property of the matrix is preserved, which makes the iterative solvers more robust. The final set of equations will consist of N_D identity equations and (N - N_D) finite element equations for N unknowns S_i's. After S_i's for all nodes are solved from the matrix equation, Eq. (3.1.146) is then used to back calculate N_D Ψ_I^B's.

If a direct solver is used to solve the matrix equation, the above procedure will solve N S_i's accurately except for roundoff errors. However, if an iterative solver is used, stopping criteria must be strict enough so that the converged solution of N S_i's are accurate enough to the exact solution. With such accurate S_is, then can be sure that the back-calculated ND Ψ_I^B 's are accurate.

Cauchy boundary condition: prescribed salt flux

If Ψ_I^B is given (Cauchy flux boundary condition), all coefficients ($C_{I,I-1}$, $C_{I,I}$, $C_{I,I+1}$) and right-hand side (L_I, Ψ_I^i, Ψ_I^{o1}, Ψ_I^{o2}) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then Eq. (3.1.146) is modified to incorporate the boundary conditions and used to solve for S_I. The modification of Eq. (3.1.146) is straightforward. Because Ψ_I^B is a known quantity, it contributes to the load on the right hand side. This type of boundary conditions is very easy to implement. After S_is are obtained, the original Eq. (3.1.146), which is stored in a temporary array, isused to back calculate N_C Ψ_I^B's on flux boundaries (where N_C is the number of flux boundary nodes). These back-calculated Ψ_I^B's should be theoretically identical to the input Ψ_I^B's. However, because of round-off errors (in the case of direct solvers) or because of stopping criteria (in the case of iterative solvers), the back-calculated will be slightly different from the input Ψ_I^B's. If the differences between the two are significant, it is an indication that the solvers have not yielded accurate solutions.

Neumann boundary condition: prescribed gradient of salinity

At Neumann boundaries, the temperature gradient is prescribed, thus, the flux due to temperature gradient is given. For this case, all coefficients ($C_{I,I-1}$, $C_{I,I}$, $C_{I,I+1}$) and right-hand side (L_I, Ψ_I^i, Ψ_I^{o1}, Ψ_I^{o2}) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then Eq. (3.1.146) is modified to incorporate the boundary conditions and used to solve for S_I. For the Neumann boundary condition, Ψ_I^B contributes to both the matrix coefficient and load vector, thus both the coefficient matrix [C] and the load vector {L} must be modified. Recall

$$\Psi_i^B = \left(W_i QS - N_i D^S A \frac{\partial S}{\partial x} \right) \Big|_{x=X_1}^{x=X_N} \tag{3.1.148}$$

Apply this equation to Node I, we have

$$\Psi_I^B \equiv n_I Q S_I - n_I D^S A \frac{\partial S}{\partial x} \Big|_{x=X_I} = n_I Q S_I - \Psi_I^{nb} \tag{3.1.149}$$

where n_I is the unit outward normal vector at the boundary node I, Ψ_I^{nb} is the Neumann boundary flux at node I. Substitution of Eq. (3.1.149) into Eq. (3.1.146), we have the modified coefficient matrix and load vector; thus the modified Eq. (3.1.146). This modified equation is used to solve S_I. After S_I is solved, the original Eq. (3.1.146) (recall the original Eq. (3.1.146) must be and has been stored in a temporary array) is used to back-calculate Ψ_I^B.

Variable boundary condition:

At the variable boundary condition Node I, the implementation of boundary conditions can be made identical to that for a Cauchy boundary condition node if the flow is directed into the river/stream/canal reach. If the flow is going out of the reach, the boundary condition is implemented similar to the implementation of Neuman boundary condition with $\Psi_I^{nb} = 0$. The assumption of zero Neumann flux implies that a Neuman node must be far away from the source/sink.

Junction boundary condition:

If the node IJ is an internal node that connects a junction J, then node IJ is treated as an internal boundary node. For example, consider three reaches with three internal nodes connecting to the junction J (Fig. 3.1-8). After applying the finite element method to Eq. (3.1.139), we have a total of (1J + 2J + 3J) algebraic equations. The algebraic equations for Nodes 1J, 2J, and 3J can be written based on Eq. (3.1.146)

$$C_{1J,1J-1}^1 S_{1J-1}^1 + C_{1J,1J}^1 S_{1J}^1 = L_{1J}^1 + \left(\Psi_{1J}^{i1} + \Psi_{1J}^{o11} + \Psi_{1J}^{o21} \right) - \Psi_{1J}^1 \tag{3.1.150}$$

$$C_{2J,2J-1}^2 S_{2J-1}^2 + C_{2J,2J}^2 S_{2J}^2 = L_{2J}^2 + \left(\Psi_{2J}^{i2} + \Psi_{2J}^{o12} + \Psi_{2J}^{o22} \right) - \Psi_{2J}^2 \tag{3.1.151}$$

$$C_{3J,3J-1}^3 S_{3J-1}^3 + C_{3J,3J}^3 S_{3J}^3 = L_{3J}^3 + \left(\Psi_{3J}^{i3} + \Psi_{3J}^{o13} + \Psi_{3J}^{o23} \right) - \Psi_{1J}^3 \tag{3.1.152}$$

where the superscript denotes the reach number and subscript denotes local node number in a reach, for example, S_{1J}^1 denotes the salinity at the 1J-th node in Reach 1. For a convenient discussion, let us associate each of the unknowns, S_1^1, …, S_{1J-1}^1 to each of the 1J-1 finite element equations in Reach 1. Similarly, we associate each of the unknowns, S_1^2, …, S_{2J-2}^2 to each of the 2J-1 finite element equations in Reach 2 and each of the unknowns and S_1^3, …, S_{3J-1}^3 to each of the 3J-1 finite element equations in Reach 3. The unknowns, Ψ_{1J}^1, Ψ_{2J}^2, and Ψ_{3J}^3, are absent from these (1J-1 + 2J-1 + 3J-1) equations. In other words, we can say each equation governs one unknown. However, two unknowns, S_{1J}^1 and Ψ_{1J}^1, appear in Equation (3.1.150). Similarly, Equation (3.1.151) has two unknowns, S_{2J}^2 and Ψ_{2J}^2, and Equation (3.1.152) has two unknowns, S_{3J}^3 and Ψ_{3J}^3. The number of unknowns, (1J + 2J + 3J) salinities and Ψ_{1J}^1, Ψ_{2J}^2, and Ψ_{3J}^3, is more than the number of equations, (1J + 2J + 3J) finite element equations. Three more governing equations must be set up, which can be obtained with the assumption that the salt flux is due mainly to advection as

$$\Psi_{1J}^1 \equiv \left(QS - D^S A \frac{\partial S}{\partial x} \right)\Bigg|_{1J} = \frac{1}{2} Q_{1J}^1 \left[\left(1 + sign\left(Q_{1J}^1\right)\right) S_{1J}^1 + \left(1 - sign\left(Q_{1J}^1\right)\right) S_J \right] \qquad \textbf{(3.1.153)}$$

$$\Psi_{2J}^2 \equiv \left(QS - D^S A \frac{\partial S}{\partial x} \right)\Bigg|_{2J} = \frac{1}{2} Q_{2J}^2 \left[\left(1 + sign\left(Q_{2J}^2\right)\right) S_{2J}^2 + \left(1 - sign\left(Q_{2J}^2\right)\right) S_J \right] \qquad \textbf{(3.1.154)}$$

$$\Psi_{3J}^3 \equiv \left(QS - D^S A \frac{\partial S}{\partial x} \right)\Bigg|_{3J} = \frac{1}{2} Q_{3J}^3 \left[\left(1 + sign\left(Q_{3J}^3\right)\right) S_{3J}^3 + \left(1 - sign\left(Q_{3J}^3\right)\right) S_J \right] \qquad \textbf{(3.1.155)}$$

where Q_{1J}^1, Q_{2J}^2, and Q_{3J}^3, respectively, are the volumetric flow rates from/to Nodes 1J, 2J, and 3J, respectively, to/from the junction J [cf. Eqs. (3.1.71), (3.1.72), and (3.1.73), respectively].

Equations (3.1.153) through (3.1.155) introduce one additional unknown, S_J. One additional equation must be set up which can be done based on the energy budget at the junction J. The rate of change of energy at the junction J must be equal to the net energy rate from all reaches that join at J. This energy budget can be written as

$$\frac{d(V_J S_J)}{dt} = \sum_i \Psi_{iJ}^i \qquad \textbf{(3.1.156)}$$

When the storage effect of the junction is small, the salt budget Eq. (3.1.156) is not employed. Instead, the following equation, resulting from the requirement that the summation of salt flux is equal to zero, is used

$$\sum_{i=1}^{i=3} \Psi_{iJ}^i = 0 \qquad \textbf{(3.1.157)}$$

Equations (3.1.150) through (3.1.155) and Eq. (3.1.156) or Eq. (3.1.157) constitute 7 equations for seven unknowns, S_{1J}^1, S_{2J}^2, S_{3J}^3, Ψ_{1J}^1, Ψ_{2J}^2, Ψ_{3J}^3, and S_J. If there are N_J junctions, there will be N_J blocks of seven equations. These N_J blocks of equations should be solved iteratively along with N_R block of finite element equations where N_R is the number of reaches. In other words, the whole system of algebraic equations can be solved with block iterations. Each block of equations can be solved directly. For example, each of N_R blocks of finite element equations can be solved with an efficient tri-diagonal matrix solver such as the Thomas algorithm. Each of the N_J blocks of seven equations can be solved with the Gaussian direct elimination with full pivoting.

Control structure boundary condition:

The control structures may include weirs, gates, culverts, etc. For the two internal boundary nodes 1S and 2S separated by a structure, the boundary conditions at these two nodes are given by

$$\Psi_{1S} = \left(QS - D^S A \frac{\partial S}{\partial x} \right)\Bigg|_{1S} = \frac{1}{2} Q\left[\left(1 + sign(Q)\right) S_{1S} + \left(1 - sign(Q)\right) S_{2S} \right] \qquad \textbf{(3.1.158)}$$

$$\Psi_{2S} = \left(QS - D^S A \frac{\partial S}{\partial x} \right)\Bigg|_{2S} = \frac{1}{2} Q\left[\left(1 + sign(Q)\right) S_{2S} + \left(1 - sign(Q)\right) S_{2S} \right] \qquad \textbf{(3.1.159)}$$

where Ψ_{1S} is the salt flux through node 1S; Φ_{2s} is the salt flux through node 2S; and Q is the flow rate through the structure S; sign(Q) is equal 1.0 if the flow is from node 1S to node 2S, -1.0 if flow is from node 2S to node 1S; S_{1S} is the temperature at node 1S; and S_{2S} is the temperature at node 2S.

3.1.5.2 The Hybrid Lagrangian-Eulerian Finite Element Method. When the hybrid Lagrangian-Eulerian finite element method is used to solve the salt transport equation, we expand Eq. (3.1.139) to yield following advection-dispersion equation in the Lagrangian form

$$\frac{D_V S}{Dt} + KS = D + \Psi^S + \Psi^I + \Psi^{O1} + \Psi^{O2} \quad where \quad V = \frac{Q}{A} \qquad \textbf{(3.1.160)}$$

in which

$$K = \frac{1}{A}\frac{\partial A}{\partial t} + \frac{1}{A}\frac{\partial Q}{\partial x}, \quad D = \frac{1}{A}\frac{\partial}{\partial x}\left(D^S A \frac{\partial S}{\partial x} \right)$$
$$\Psi^S = \frac{M_s^a + M_s^r}{A}, \quad and \quad \Psi^I = \frac{M_s^i}{A}, \Psi^{O1} = \frac{M_s^{o1}}{A}, \Psi^{O2} = \frac{M_s^{O2}}{A} \qquad \textbf{(3.1.161)}$$

To use the semi-Lagrangian method to solve the thermal transport equation, we integrate Eq. (3.1.160) along its characteristic line from x_i at new time level to x_i^* (Fig. 3.1-12), we obtain

$$\begin{aligned} \left(1 + \frac{\Delta\tau}{2} K_i^{(n+1)}\right) S_i^{(n+1)} &= \left(1 - \frac{\Delta\tau}{2} K_i^*\right) S_i^* + \frac{\Delta\tau}{2}\left(D_i^{(n+1)} + D_i^*\right) + \frac{\Delta\tau}{2}\left(\Psi_i^{S^{(n+1)}} + \Psi_i^{S*}\right) \\ &+ \frac{\Delta\tau}{2}\left(\Psi_i^{I^{(n+1)}} + \Psi_i^{I*}\right) + \frac{\Delta\tau}{2}\left(\Psi_i^{O1^{(n+1)}} + \Psi_i^{O1*}\right) + \frac{\Delta\tau}{2}\left(\Psi_i^{O2^{(n+1)}} + \Psi_i^{O2*}\right), \quad i \in N \end{aligned} \qquad \textbf{(3.1.162)}$$

where $\Delta\tau$ is the tracking time, it is equal to Δt when the backward tracking is carried out all the way to the root of the characteristic and it is less than Δt when the backward tracking hits the boundary before Δt is consumed; $K_i^{(n+1)}$, $S_i^{(n+1)}$, $D_i^{(n+1)}$, $\Psi_i^{S(n+1)}$, $\Psi_i^{I(n+1)}$, $\Psi_i^{O1(n+1)}$, and $\Psi_i^{O2(n+1)}$ respectively, are the values of K, S, D, Ψ^S, Ψ^I, Ψ^{O1}, and Ψ^{O2}, respectively, at x_i at new time level $t = (n+1)\Delta t$; and K_i^*, S_i^*, D_i^*, Ψ_i^{S*}, Ψ_i^{I*}, Ψ_i^{O1*}, and Ψ_i^{O2*}, respectively, are the values of K, S, D, Ψ^S, Ψ^I, Ψ^{O1}, and Ψ^{O2}, respectively, at the location x_i^*.

To compute the dispersion/diffusion terms $D_i^{(n+1)}$ and D_i^*, we rewrite the second equation in Eq. (3.1.161) as

$$AD = \frac{\partial}{\partial x}\left(D^S A \frac{\partial S}{\partial x} \right) \qquad \textbf{(3.1.163)}$$

Applying the finite element method to Eq. (3.1.163) at new time level (n+1), we obtain the following matrix equation for $\{D^{(n+1)}\}$ as

$$\left[a^{(n+1)}\right]\left\{D^{(n+1)}\right\} + \left[b^{(n+1)}\right]\left\{S^{(n+1)}\right\} = \left\{B^{(n+1)}\right\} \qquad \textbf{(3.1.164)}$$

in which

$$\left\{D^{(n+1)}\right\} = \left\{D_1^{(n+1)} \quad D_2^{(n+1)} \quad . \quad . \quad D_i^{(n+1)} \quad . \quad . \quad D_N^{(n+1)}\right\}^{Transpose} \qquad \textbf{(3.1.165)}$$

$$\left\{S^{(n+1)}\right\} = \left\{S_1^{(n+1)} \quad S_2^{(n+1)} \quad . \quad . \quad S_i^{(n+1)} \quad . \quad . \quad S_N^{(n+1)}\right\}^{Transpose} \qquad \textbf{(3.1.166)}$$

$$\left\{B^{(n+1)}\right\} = \left\{B_1^{(n+1)} \quad B_2^{(n+1)} \quad . \quad . \quad B_i^{(n+1)} \quad . \quad . \quad B_N^{(n+1)}\right\}^{Transpose} \qquad \textbf{(3.1.167)}$$

$$a_{ij}^{(n+1)} = \int_{X_1}^{X_N} N_i A\Big|_{(n+1)} N_j dx, \quad b_{ij}^{(n+1)} = \int_{X_1}^{X_N} \frac{dN_i}{dx}\left(D^S A\right)\Big|_{(n+1)} \frac{dN_j}{dx} dx,$$
$$B_i^{(n+1)} = nN_i\left(D^S A\right)\Big|_{(n+1)} \frac{\partial S^{(n+1)}}{\partial x}\Bigg|_{x=x_1}^{x=x_N} \qquad \textbf{(3.1.168)}$$

where the superscript (n+1) denotes the time level; N_i and N_j are the base functions of nodes at x_i and x_j, respectively.

Lumping the matrix $[a^{(n+1)}]$, we can solve Eq. (3.1.164) for $D_I^{(n+1)}$ as follows

$$D_I^{(n+1)} = -\frac{1}{a_{II}^{(n+1)}}\sum_j b_{Ij}^{(n+1)} S_j^{(n+1)} \quad \text{if} \quad I \in \{2,3,..,N-1\}$$
$$D_I^{(n+1)} = \frac{1}{a_{II}^{(n+1)}} B_I^{(n+1)} - \frac{1}{a_{II}^{(n+1)}}\sum_j b_{Ij}^{(n+1)} S_j^{(n+1)} \quad \text{if} \quad I \in \{1,N\} \qquad \textbf{(3.1.169)}$$

where $a_{II}^{(n+1)}$ is the lumped $a_{ii}^{(n+1)}$. Following the identical procedure that leads Eq. (3.1.163) to Eq. (3.1.169), we have

$$D_I^{(n)} = -\frac{1}{a_{II}^{(n)}}\sum_j b_{Ij}^{(n)} S_j^{(n)} \quad \text{if} \quad I \in \{2,3,..,N-1\}$$
$$D_I^{(n)} = \frac{1}{a_{II}^{(n)}} B_I^{(n)} - \frac{1}{a_{II}^{(n)}}\sum_j b_{Ij}^{(n)} S_j^{(n)} \quad \text{if} \quad I \in \{1,N\} \qquad \textbf{(3.1.170)}$$

where $\{B^{(n)}\}$, $\{a^{(n)}\}$ and $\{b^{(n)}\}$, respectively, are defined similar to $\{B^{(n+1)}\}$, $\{a^{(n+1)}\}$ and $\{b^{(n+1)}\}$, respectively.

With $\{D^{(n)}\}$ calculated with Eq. (3.1.170), $\{D^*\}$ can be interpolated. Substituting Eq. (3.1.169) into Eq. (3.1.162) and implementing boundary conditions given in Section 2.1.4, we obtain a system of N simultaneous algebraic equations N unknowns ($S_i^{(n+1)}$ for i = 1, 2, .., N.) If the dispersion/diffusion term is not included, then Eq. (3.1.162) is reduced to a set of N decoupled equations as

$$a_{ii} S_i^{(n+1)} = b_i, \quad i \in N \tag{3.1.171}$$

where

$$a_{ii} = \left(1 + \frac{\Delta\tau}{2} K_i^{(n+1)} \right) \tag{3.1.172}$$

$$\begin{aligned} b_i = {} & \left(1 - \frac{\Delta\tau}{2} K_i^* \right) S_i^* + \frac{\Delta\tau}{2}\left(\Psi_i^{S^{(n+1)}} + \Psi_i^{S*} \right) + \frac{\Delta\tau}{2}\left(\Psi_i^{I^{(n+1)}} + \Psi_i^{1*} \right) \\ & + \frac{\Delta\tau}{2}\left(\Psi_i^{O1^{(n+1)}} + \Psi_i^{O1*} \right) + \frac{\Delta\tau}{2}\left(\Psi_i^{O2^{(n+1)}} + \Psi_i^{O2*} \right), \quad i \in N \end{aligned} \tag{3.1.173}$$

Equation (3.1.171) is applied to all interior nodes without having to make any modification. On a boundary point, there are two possibilities: Eq. (3.1.171) is replaced with a boundary equation when the flow is directed into the reach or Eq. (3.1.171) is still valid when the flow is direct out of the reach. In other words, when the salt is transported out of the region at a boundary node (i.e., when $\mathbf{N}\bullet\mathbf{V} \geq 0$), a boundary condition is not needed and Equation (3.1.171) is used to compute the $S_i^{(n+1)}$. When the salt is transported into the region at a node (i.e., when $\mathbf{N}\bullet\mathbf{V} < 0$), a boundary condition must be specified.

Alternatively, to facilitate the implementation of boundary condition at incoming flow node, the algebraic equation for the boundary node is obtained by applying the finite element method to the boundary node rather than the use of particle tracking. For this alternative approach, the implementation of boundary conditions at global boundary nodes, internal junction nodes, and internal nodes connecting to control structures is identical to that in the finite element approximation of solving the salt transport equation.

3.2 Solving the Two-Dimensional Overland Flow Equations

As in solving the one-dimensional flow equations for river/stream/canal networks, we employ a variety of numerical approaches to solve two-dimensional overland flow equations. For fully dynamic wave models, we cast the governing equations in characteristic forms and solve the governing equations with the hybrid Lagrangian-Eulerian finite element method. For diffusive wave models, we use either the conventional finite element methods or hybrid Lagrangian-Eulerian finite element methods. For kinematic wave models, we use semi-Lagrangian methods.

3.2.1 The Lagrangian-Eulerian Finite Element Method for Dynamic Waves

To facilitate the application of hybrid Lagrangian-Eulerian finite element method to fully dynamic wave models, substituting A_1, A_2, A_3, B_1, B_2, and B_3 in Eq. (2.2.27); R_1, R_2, and R_3 in Eq. (2.2.9); and

D_x and D_y in (2.2.10) into Eqs. (2.2.28) through and (2.2.30), and rearranging the resulting equations, we obtain

$$k_y^{(1)}\frac{D_V u}{D\tau} - k_x^{(1)}\frac{D_V v}{D\tau} + S_1 = D_\otimes - k_y^{(1)}Ku + k_x^{(1)}Kv + S_\otimes \quad \textbf{(3.2.1)}$$

$$2\frac{D_{V+ck^{(2)}}c}{D\tau} + k_x^{(2)}\frac{D_{V+ck^{(2)}}u}{D\tau} + k_y^{(2)}\frac{D_{V+ck^{(2)}}v}{D\tau} + S_2 = D_\pm - k_x^{(2)}Ku - k_y^{(2)}Kv + S_+ \quad \textbf{(3.2.2)}$$

$$-2\frac{D_{V-ck^{(2)}}c}{D\tau} + k_x^{(2)}\frac{D_{v-ck^{(2)}}u}{D\tau} + k_y^{(2)}\frac{D_{V-ck^{(2)}}v}{D\tau} + S_3 = D_\pm - k_X^{(2)}Ku - k_y^{(2)}Kv + S_- \quad \textbf{(3.2.3)}$$

in which

$$D_\otimes = k_y^{(1)}D_x - k_x^{(1)}D_y, \quad D_\pm = k_x^{(2)}D_x + k_y^{(2)}D_y, \quad and \quad K = \frac{S_S + S_R - S_E + S_I}{h} + \frac{\kappa|V|}{h} \quad \textbf{(3.2.4)}$$

$$D_x = \frac{1}{h}\left[\frac{\partial}{\partial x}\left(h\varepsilon_{xx}\frac{\partial u}{\partial x}\right) + \frac{\partial}{\partial y}\left(h\varepsilon_{xy}\frac{\partial u}{\partial y} + h\varepsilon_{yx}\frac{\partial v}{\partial x}\right)\right] \quad \textbf{(3.2.5)}$$

$$D_y = \frac{1}{h}\left[\frac{\partial}{\partial x}\left(h\varepsilon_{xy}\frac{\partial u}{\partial y} + h\varepsilon_{yx}\frac{\partial v}{\partial x}\right) + \frac{\partial}{\partial y}\left(h\varepsilon_{yy}\frac{\partial v}{\partial y}\right)\right] \quad \textbf{(3.2.6)}$$

$$\begin{aligned} S_\otimes = k_y^{(1)}&\left(-g\frac{\partial Z_o}{\partial x} + \frac{\left(M_x^S + M_x^R - M_x^E + M_x^I\right)}{h} + \frac{\tau_x^s}{\rho h}\right) \\ &- k_x^{(1)}\left(-g\frac{\partial Z_o}{\partial y} + \frac{\left(M_y^S + M_y^R - M_y^E + M_y^I\right)}{h} + \frac{\tau_y^s}{\rho h}\right) \end{aligned} \quad \textbf{(3.2.7)}$$

$$\begin{aligned} S_+ = \frac{g}{c}\left(S_S + S_R - S_E + S_I\right) &+ k_x^{(2)}\left(-g\frac{\partial Z_o}{\partial x} + \frac{\left(M_x^S + M_x^R - M_x^E + M_x^I\right)}{h} + \frac{\tau^S}{\rho h}\right) \\ &+ k_y^{(2)}\left(-g\frac{\partial Z_o}{\partial y} + \frac{\left(M_y^S + M_y^R - M_y^E + M_y^I\right)}{h} + \frac{\tau_y^s}{\rho h}\right) \end{aligned} \quad \textbf{(3.2.8)}$$

$$\begin{aligned} S_- = -\frac{g}{c}\left(S_S + S_R - S_E + S_I\right) &+ k_x^{(2)}\left(-g\frac{\partial Z_o}{\partial x} + \frac{M_x^S + M_x^R - M_x^E + M_x^I}{h} + \frac{\tau_X^s}{\rho h}\right) \\ &+ k_y^{(2)}\left(-g\frac{\partial Z_o}{\partial y} + \frac{\left(M_y^S + M_y^R - M_y^E + M_y^I\right)}{h} + \frac{\tau_y^s}{\rho h}\right) \end{aligned} \quad \textbf{(3.2.9)}$$

where $D_\otimes$ is the diffusive transport of the vorticity wave; $D_\pm$ is the diffusive transport of the positive and negative gravity waves; K is the decay coefficient for all three waves; and $S_\otimes$, S_+, and S_- are the sources/sinks of the vorticity, positive, and negative waves, respectively.

Integrating Eqs. (3.2.1) through (3.2.3) along their respective characteristic lines from x to x_1*, x_2*, and x_3* (Fig. 3.2-1), we obtain

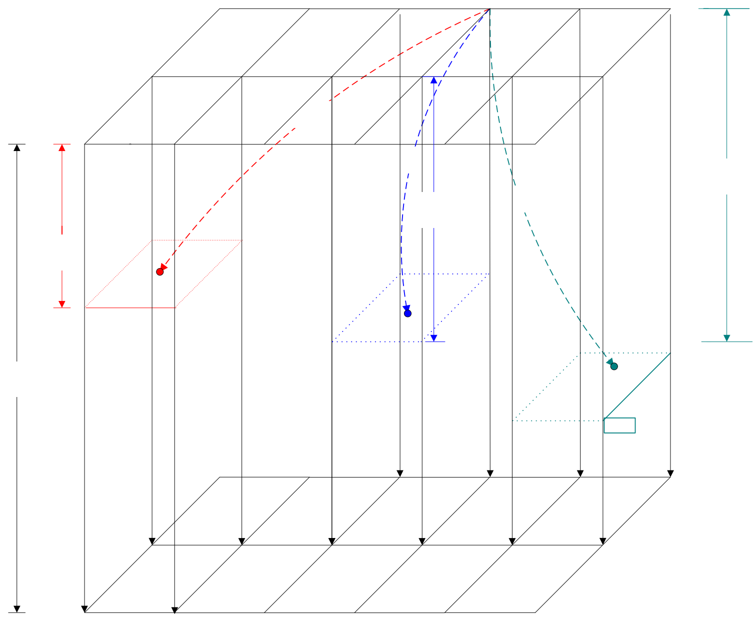

Fig. 3.2-1. Backward Particle Tracking along Characteristic Lines in Two Dimensions.

$$\overline{k_y^{(1)}}\frac{u-u_1^*}{\Delta\tau_1}-\overline{k_x^{(1)}}\frac{v-v_1^*}{\Delta\tau_1}+\frac{1}{2}\left(S_1+\left(S_1\right)_1^*\right)$$
$$=\frac{1}{2}\left(D_\otimes+\left(D_\otimes\right)_1^*\right)-\frac{1}{2}\left(k_y^{(1)}Ku-k_x^{(1)}Kv+\left(k_y^{(1)}Ku\right)_1^*-\left(k_x^{(1)}Kv\right)_1^*\right)+\frac{1}{2}\left(S_\otimes+\left(S_\otimes\right)_1^*\right) \quad \textbf{(3.2.10)}$$

t

$$\frac{2c-2c_2^*}{\Delta\tau_2}+\overline{k_x^{(2)}}\frac{u-u_2^*}{\Delta\tau_2}+\overline{k_y^{(2)}}\frac{v-v_2^*}{\Delta\tau_2}+\frac{1}{2}\left(S_2+\left(S_2\right)_2^*\right)$$
$$=\frac{1}{2}\left(D_\pm+\left(D_\pm\right)_2^*\right)-\frac{1}{2}\left(k_x^{(2)}Ku+k_y^{(2)}Kv+\left(k_x^{(2)}Ku\right)_2^*+\left(k_y^{(2)}Kv\right)_2^*\right)+\frac{1}{2}\left(S_++\left(S_+\right)_2^*\right) \quad \textbf{(3.2.11)}$$

$$-\frac{2c-2c_3^*}{\Delta\tau_3}+\overline{k_x^{(2)}}\frac{u-u_3^*}{\Delta\tau_3}+\overline{k_y^{(2)}}\frac{v-v_3^*}{\Delta\tau_3}+\frac{1}{2}\left(S_3+\left(S_3\right)_3^*\right)$$
$$=\frac{1}{2}\left(D_\pm+\left(D_\pm\right)_3^*\right)-\frac{1}{2}\left(k_x^{(2)}Ku+k_y^{(2)}Kv+\left(k_x^{(2)}Ku\right)_3^*+\left(k_y^{(2)}Kv\right)_3^*\right)+\frac{1}{2}\left(S_-+\left(S_-\right)_3^*\right) \quad \textbf{(3.2.12)}$$

$\Delta\tau_2$

where u_1^*, v_1^*, and $\Delta\tau_1$ are determined by backward tracking along the first characteristic; c_2^*, u_2^*, v_2^*, and $\Delta\tau_2$ are determined by backward tracking along the second characteristic; c_3^*, u_3^*, v_3^*, and $\Delta\tau_3$ are determined by backward tracking along the third characteristic; and all other variables with a superscript * are determined similarly at the roots of particle tracking.

j1

Δt

In Eqs. (3.2.11) through (3.2.13), the primitive variables at the backward tracked locations are interpolated with those at the global nodes and at both new and old time levels as

$$c_1^* = a_1\, c_{k1}^n + a_2\, c_{k2}^n + a_3\, c_{k3}^n + a_4\, c_{k4}^n + a_5\, c_{k1} + a_6\, c_{k2} + a_7\, c_{k3} + a_8\, c_{k4} \tag{3.2.13}$$

$$u_1^* = a_1\, u_{k1}^n + a_2\, u_{k2}^n + a_3\, u_{k3}^n + a_4\, u_{k4}^n + a_5\, u_{k1} + a_6\, u_{k2} + a_7\, u_{k3} + a_8\, u_{k4} \tag{3.2.14}$$

$$v_1^* = a_1\, v_{k1}^n + a_2\, u_{k2}^n + a_3\, v_{k3}^n + a_4\, v_{k4}^n + a_5\, v_{k1} + a_6\, v_{k2} + a_7\, v_{k3} + a_8\, v_{k4} \tag{3.2.15}$$

$$c_2^* = b_1\, c_{j1}^n + b_2\, c_{j2}^n + b_3\, c_{j3}^n + b_4\, c_{j4}^n + b_5\, c_{j1} + b_6\, cj + b_7\, c_{j3} + b_8\, c_{j4} \tag{3.2.16}$$

$$u_2^* = b_1\, u_{j1}^n + b_2\, u_{j2}^n + b_3\, u_{j3}^n + b_4\, u_{j4}^n + b_5\, c_{j1} + b_6\, u_{j2} + b_7\, u_{j3} + b_8\, u_{j4} \tag{3.2.17}$$

$$v_2^* = b_1\, v_{j1}^n + b_2\, v_{j2}^n + b_3\, v_{j3}^n + b_4\, v_{j4}^n + b_5\, v_{j1} + b_6\, v_{j2} + b_7\, v_{j3} + b_8\, v_{j4} \tag{3.2.18}$$

$$c_3^* = d_1\, c_{m1}^n + d_2\, c_{m2}^n + d_3\, c_{m3}^n + d_4\, c_{m4}^n + d_5\, c_{m1} + d_6\, c_{m2} + d_7\, c_{m3} + d_8\, c_{m4} \tag{3.2.19}$$

$$u_3^* = d_1\, u_{m1}^n + d_2\, u_{m2}^n + d_3\, u_{m3}^n + d_4\, {}_{m4}^n + d_5\, u_{m1} + d_6\, {}_{m2} + d_7\, u_{m3} + d_8\, u_{m4} \tag{3.2.20}$$

$$v_3^* = d_1\, v_{m1}^n + d_2\, v_{m2}^n + d_3\, v_{m3}^n + d_4\, v_{m4}^n + d_5\, v_{mk1} + d_6\, v_{m2} + d_7\, v_{m3} + d_8\, v_{m4} \tag{3.2.21}$$

where a_1 through a_8, b_1 through b_8, and d_1 through d_8 are interpolation parameters, all in the ranges of [0,1]; $k1$, $k2$, $k3$, and $k4$ are nodes of the element that the backward tracking, along the first characteristic, stops at; $j1$, $j2$, $j3$, and $j4$ are nodes of the element that the backward tracking, along the second characteristic, stops at; $m1$, $m2$, $m3$, and $m4$ are nodes of the element that the backward tracking, along the third characteristic, stops at (Fig. 3.2-1). It should be noted that we may use two given parameters to determine where to stop in the backward tracking: one is for controlling tracking time and the other one is for controlling tracking distance. After the primitive variables at the backward tracked points are interpolated, all other parameters (such as the decay coefficients and source/sink terms) are functions of these variables and can be calculated.

To calculate D_x and D_y, we multiple Eqs. (3.2.5) and (3.2.6) by h to yield

$$hD_x = \frac{\partial}{\partial x}\left(h\varepsilon_{xx} \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y}\left(h\varepsilon_{xy} \frac{\partial u}{\partial y} + h\varepsilon_{yx} \frac{\partial v}{\partial x} \right) \tag{3.2.22}$$

$$hD_y = \frac{\partial}{\partial x}\left(h\varepsilon_{xy} \frac{\partial u}{\partial y} + h\varepsilon_{yx} \frac{\partial v}{\partial x} \right) + \frac{\partial}{\partial y}\left(h\varepsilon_{yy} \frac{\partial v}{\partial y} \right) \tag{3.2.23}$$

Applying the Galerkin finite element method to Eqs. (3.2.22) and (3.2.23), we obtain the following matrix equations for D_x and D_y

$$[QA]\{D_x\} + [QB]\{u\} + [QC]\{v\} = \{F_x\} \tag{3.2.24}$$

$$[QA]\{D_y\}+[QD]\{u\}+[QE]\{v\}=\{F_y\} \qquad \textbf{(3.2.25)}$$

where

$$QA_{ij}=\int_R N_i h N_j dR \qquad \textbf{(3.2.26)}$$

$$QB_{ij}=\int_R \nabla N_i^* \begin{bmatrix} h\varepsilon_{xx} & 0 \\ 0 & h\varepsilon_{xy} \end{bmatrix} \cdot \nabla N_j dR; \quad QC_{ij}=\int_R \nabla N_i^* \begin{bmatrix} 0 & 0 \\ h\varepsilon_{xy} & 0 \end{bmatrix} \cdot \nabla N_j dR \qquad \textbf{(3.2.27)}$$

$$QD_{ij}=\int_R \nabla N_i^* \begin{bmatrix} 0 & h\varepsilon_{xy} \\ 0 & 0 \end{bmatrix} \cdot \nabla N_j dR; \quad QE_{ij}=\int_R \nabla N_i^* \begin{bmatrix} h\varepsilon_{xy} & 0 \\ 0 & h\varepsilon_{yy} \end{bmatrix} \cdot \nabla N_j dR \qquad \textbf{(3.2.28)}$$

$$F_{xi}=\sum_{e\in M_e}\int_{B_e}\left\{ N_a^e \mathbf{n} \cdot \begin{bmatrix} h\varepsilon_{xx} & 0 \\ 0 & h\varepsilon_{xy} \end{bmatrix} \cdot \nabla u + N_a^e \mathbf{n} \cdot \begin{bmatrix} 0 & 0 \\ h\varepsilon_{xy} & 0 \end{bmatrix} \cdot \nabla v \right\} dB \qquad \textbf{(3.2.29)}$$

$$F_{yi}=\sum_{e\in M_e}\int_{B_e}\left\{ N_a^e \mathbf{n} \cdot \begin{bmatrix} 0 & h\varepsilon_{xy} \\ 0 & 0 \end{bmatrix} \cdot \nabla u + N_a^e \mathbf{n} \cdot \begin{bmatrix} h\varepsilon_{xy} & 0 \\ 0 & h\varepsilon_{yy} \end{bmatrix} \cdot \nabla v \right\} dB \qquad \textbf{(3.2.30)}$$

Lumping the matrix [QA], we can explicitly compute $\{D_x\}$ and $\{D_y\}$ in terms of {u} and {v}.

$$D_{xi}=\frac{1}{QA_{ii}}F_{xi}-\frac{1}{QA_{ii}}\sum_j QB_{ij}u_j-\frac{1}{QA_{ii}}\sum_j QC_{ij}v_j \qquad \textbf{(3.2.31)}$$

and

$$D_{yi}=\frac{1}{QA_{ii}}F_{yi}-\frac{1}{QA_{ii}}\sum_j QD_{ij}u_j-\frac{1}{QA_{ii}}\sum_j QE_{ij}v_j \qquad \textbf{(3.2.32)}$$

Following the identical procedure that leads Eqs. (3.2.22) and (3.2.23) to Eqs. (3.2.31) and (2.3.32), we have

$$D_{xi}^{(n)}=\frac{1}{QA_{ii}^{(n)}}F_{xi}^{(n)}-\frac{1}{QA_{ii}^{(n)}}\sum_j QB_{ij}^{(n)}u_j^{(n)}-\frac{1}{QA_{ii}^{(n)}}\sum_j QC_{ij}^{(n)}v_j^{(n)} \qquad \textbf{(3.2.33)}$$

and

$$D_{yi}^{(n)}=\frac{1}{QA_{ii}^{(n)}}F_{yi}^{(n)}-\frac{1}{QA_{ii}^{(n)}}\sum_j QD_{ij}^{(n)}u_j^{(n)}-\frac{1}{QA_{ii}^{(n)}}\sum_j QE_{ij}^{(n)}v_j^{(n)} \qquad \textbf{(3.2.34)}$$

where the superscript (*n*) denotes that the variables are to be evaluated at the old time level *n*.

Similar to Eqs. (3.2.13) through (3.2.21), $(D_{xi}^*)_1$, $(D_{xi}^*)_2$, and $(D_{xi}^*)_3$ and $(D_{yi}^*)_1$, $(D_{yi}^*)_2$, and $(D_{yi}^*)_3$ at the backward tracked location are interpolated with {D} and $\{D^{(n)}\}$ as

$$\left(D_{xi}^*\right)_1=a_1D_{xk1}^n+a_2D_{xk2}^n+a_3D_{xk3}^n+a_4D_{xk4}^n+a_5D_{xk1}+a_6D_{xk2}+a_7D_{xk3}+a_8D_{xk4} \qquad \textbf{(3.2.35)}$$

$$\left(D_{xi}^{*}\right)_2 = b_1 D_{xj1}^{n} + b_2 D_{xj2}^{n} + b_3 D_{xj3}^{n} + b_4 D_{xj4}^{n} + b_5 D_{xj1} + b_6 D_{xj2} + b_7 D_{xj3} + b_8 D_{xj4} \quad \textbf{(3.2.36)}$$

$$\left(D_{xi}^{*}\right)_3 = d_1 D_{xm1}^{n} + d_2 D_{xm2}^{n} + d_3 D_{xm3}^{n} + d_4 D_{xm4}^{n} + d_5 D_{xm1} + d_6 D_{xm2} + d_7 D_{xm3} + d_8 D_{xm4} \quad \textbf{(3.2.37)}$$

$$\left(D_{yi}^{*}\right)_1 = a_1 D_{yk1}^{n} + a_2 D_{yk2}^{n} + a_3 D_{yk3}^{n} + a_4 D_{yk4}^{n} + a_5 D_{yk1} + a_6 D_{yk2} + a_7 D_{yk3} + a_8 D_{yk4} \quad \textbf{(3.2.38)}$$

$$\left(D_{yi}^{*}\right)_2 = b_1 D_{yj1}^{n} + b_2 D_{yj2}^{n} + b_3 D_{yj3}^{n} + b_4 D_{yj4}^{n} + b_5 D_{yj1} + b_6 D_{yj2} + b_7 D_{yj3} + b_8 D_{yj4} \quad \textbf{(3.2.39)}$$

$$\left(D_{yi}^{*}\right)_3 = d_1 D_{ym1}^{n} + d_2 D_{ym2}^{n} + d_3 D_{ym3}^{n} + d_4 D_{ym4}^{n} + d_5 D_{ym1} + d_6 D_{ym2} + d_7 D_{ym3} + d_8 D_{ym4} \quad \textbf{(3.2.40)}$$

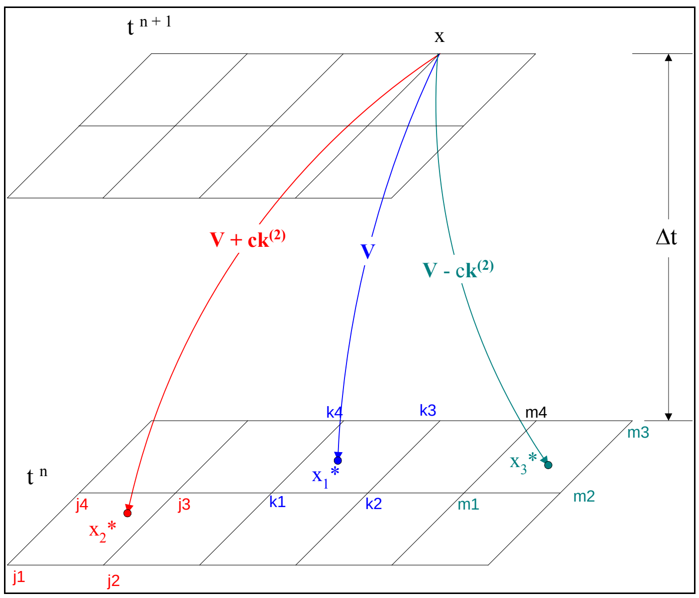

Fig. 3.2-2. Backward Tracking Along Characteristic Line to the Root in Two Dimensions

Substituting Eqs. (3.2.13) through (3.2.21) and Eqs. (3.2.35) through (3.2.40) into Eqs. (3.2.10) through (3.2.12) and implementing boundary conditions given Section 2.2.1, we obtain a system of 3N simultaneous algebraic equations for the 3N unknowns (u_i for i = 1, 2, .., N, v_i for i = 1, 2, .., N, and and c_i for i = 1,2, .., N). If the eddy diffusion terms are not included and the backward tracking is performed to reach the time level n (Fig. 3.2-2), then Eqs. (3.2.8) through (3.2.10) are reduced to a set of N decoupled triplets of equations as

$$\begin{aligned} a_{11}u + a_{12}v + a_{13}c &= B_1, \\ a_{21}u + a_{22}v + a_{23}c &= B_2, \\ a_{31}u + a_{32}v + a_{33}c &= B_3, \quad \text{for all interior nodes} \end{aligned} \quad \textbf{(3.2.41)}$$

where

$$a_{11} = \overline{k_y^{(1)}} + \frac{\Delta\tau_1}{2}\overline{(k_y^{(1)}K_+)}, \qquad a_{12} = \overline{-k_x^{(1)}} - \frac{\Delta\tau_1}{2}\overline{(k_x^{(1)}K_+)}, \qquad a_{13} = 0,$$

$$B_1 = \left(\overline{k_y^{(1)}} - \frac{\Delta\tau_1}{2}\left(k_y^{(1)}K\right)_1^*\right)u_i^* - \left(\overline{k_x^{(1)}} - \frac{\Delta\tau_1}{2}\left(k_x^{(1)}K\right)_1^*\right)v_1^* \quad \textbf{(3.2.42)}$$

$$-\frac{\tau_1}{2}\left(S_1 + (S_1)_1^*\right) + \frac{\tau_1}{2}\left(S_\oplus + (S_\oplus)_1^*\right)$$

$$a_{21} = \overline{k_x^{(2)}} + \frac{\Delta\tau_2}{2}\left(k_x^{(2)}K\right), \qquad a_{22} = \overline{k_y^{(2)}} + \frac{\Delta\tau_2}{2}\left(k_x^{(2)}K\right), \qquad a_{23} = 2,$$

$$B_2 = \left(\overline{k_x^{(2)}} - \frac{\Delta\tau_2}{2}\left(k_x^{(2)}K\right)_2^*\right)u_2^* + \left(\overline{k_y^{(2)}} - \frac{\Delta\tau_2}{2}\left(k_y^{(2)}K\right)_2^*\right)v_2^* + 2c_2^* \quad \textbf{(3.2.43)}$$

$$-\frac{\tau_2}{2}\left(S_2 + (S_2)_2^*\right) + \frac{\tau_2}{2}\left(S_+ + (S_+)_2^*\right)$$

$$a_{31} = \overline{k_x^{(2)}} + \frac{\Delta\tau_2}{2}\left(k_x^{(2)}K\right), \qquad a_{32} = \overline{k_y^{(2)}} + \frac{\Delta\tau_2}{2}\left(k_x^{(2)}K\right), \qquad a_{33} = -2,$$

$$B_3 = \left(\overline{k_x^{(2)}} - \frac{\Delta\tau_3}{2}\left(k_x^{(2)}K\right)_3^*\right)u_3^* + \left(\overline{k_y^{(2)}} - \frac{\Delta\tau_3}{2}\left(k_y^{(2)}K\right)_3^*\right)v_3^* + 2c_3^* \quad \textbf{(3.2.44)}$$

$$-\frac{\tau_3}{2}\left(S_3 + (S_3)_3^*\right) + \frac{\tau_3}{2}\left(S_+ + (S_+)_3^*\right)$$

Equations (3.2.41) is applied to all interior nodes without having to make any modification. On a boundary point, any one of the three equations in Eq. (3.2.41) must be replaced by a boundary condition equation if its corresponding wave is directed into the region from the outside world. On the other hand, if the corresponding wave is going out of the region, then the equation is valid. These conditions are addressed below for four types of physical boundaries: open upstream, open downstream, closed upstream, and closed downstream boundary nodes.

Open upstream boundary condition:

If the flow is supercritical, all three waves are directed into the region from the outside world, thus Eq. (3.2.41) is replaced with

$$\mathbf{n}\cdot\mathbf{V}h = q_n^{(up)}(t); \qquad \mathbf{n}\cdot\mathbf{V}uh + n_x\frac{gh^2}{2} = M_x^{up}; \qquad \mathbf{n}\cdot\mathbf{V}vh + n_y\frac{gh^2}{2} = M_y^{up} \quad \textbf{(3.2.45)}$$

where $\mathbf{V} = (u, v)$ is the vertically averaged velocity with u as the x-component and v the y-component; $\mathbf{n}$ is the outward unit vector normal to the boundary; q_n^{up}(t) is the flow rate of the incoming fluid from the upstream; and M_x^{up} and M_y^{up}, respectively, are the x- and y-components, respectively, of the momentum-impulse from the upstream.

If the flow is subcritical, one of the gravitational wave is going out of the region, thus Eq. (3.2.41) for the boundary point i is replaced with

$$\mathbf{n}\cdot\mathbf{V}h = q_n^{(up)}(t); \qquad \mathbf{l}\cdot\mathbf{V}h = q_\ell^{(up)}(t); \qquad a_{21}u + a_{22}v + a_{23}c = B_2$$

or **(3.2.46)**

$$\mathbf{n}\cdot\mathbf{V}h = q_n^{(up)}(t); \qquad \mathbf{l}\cdot\mathbf{V}h = q_\ell^{(up)}(t); \qquad a_{31}u + a_{32}v + a_{33}c = B_3$$

where $\mathbf{l}$ is the unit vector parallel to the boundary segment and $q_\ell^{(up)}$, a function of time t, is the flow rate parallel to the boundary.

Open downstream boundary condition:

If the flow is supercritical, all three waves are transported out of the region and Eq. (3.2.41) remains valid for the boundary point; thus

$$\begin{aligned} &a_{11}u + a_{12}v + a_{13}c = B_1, \\ &a_{21}u + a_{22}v + a_{23}c = B_2, \\ &a_{31}u + a_{32}v + a_{33}c = B_3, \quad \textit{for all interior nodes} \end{aligned} \qquad \textbf{(3.2.47)}$$

If the flow is subcritical, the vorticity wave and one the gravity waves are transported out of the region while the other gravity wave is transported into the region. Under such circumstance, Equation (3.2.41) may be replaced with

$$a_{11}u + a_{12}v + a_{13}c = B_1; \qquad a_{21}u + a_{22}v + a_{23}c_3 = B_2; \qquad h = h^{dn}(t) \;\; or \;\; \mathbf{n}\cdot\mathbf{V}h = q_n^{dn}(h)$$

or **(3.2.48)**

$$a_{11}u + a_{12}v + a_{13}c = B_1; \qquad a_{31}u + a_{32}v + a_{33}c = B_3; \qquad h = h^{dn}(t) \;\; or \;\; \mathbf{n}\cdot\mathbf{V}h = q_n^{dn}(h)$$

where $q_n^{dn}(h)$, a function of h, is the rating curve function for the downstream boundary and $h^{dn}(t)$, a function of t, is the water depth at the downstream boundary. As to which three equations in of Eq. (3.2.48) must be used depends on the physical configuration at the boundary.

Closed upstream boundary condition:

If the flow is supercritical, all three waves are transported from the boundary into the region of interest. Since neither flow nor momentum-impulse is transported from the outside world onto the boundary, the following boundary condition can be used

$$\mathbf{n}\cdot\mathbf{V}h = 0; \qquad \mathbf{n}\cdot\mathbf{V}uh + n_x\frac{gh^2}{2} = 0; \qquad \mathbf{n}\cdot\mathbf{V}vh + n_y\frac{gh^2}{2} = 0 \qquad \textbf{(3.2.49)}$$

The solution of Eq. (3.2.49) is not unique. One of the possible solution is h = 0, u = 0, and v = 0. If the flow is subcritical, one of the two gravity waves is transported out of the region, thus Equation (3.2.41) can be replaced with

$$\begin{gathered} \mathbf{n}\cdot\mathbf{V}h = 0; \qquad \mathbf{l}\cdot\mathbf{V}h = 0; \qquad a_{21}u = a_{22}v + a_{23}c_3 = B_2 \\ or \\ \mathbf{n}\cdot\mathbf{V}h = 0; \qquad \mathbf{l}\cdot\mathbf{V}h = 0; \qquad a_{31}u = a_{32}v + a_{33}c = B_3 \end{gathered} \tag{3.2.50}$$

Closed downstream boundary condition:

At the closed downstream boundary, physical condition dictates that the normal flux should be zero. In the meantime, one of the gravity wave is transported out of the region. Thus, the water depth and velocity on the boundary are determined by the internal flow dynamics and the condition of zero normal flux. The boundary condition can be stated as

$$\begin{gathered} a_{11}u + a_{12}v + a_{13}c = B_1; \qquad a_{21}u + a_{22}v + a_{23}c_3 = B_2; \qquad \mathbf{n}\cdot\mathbf{V}h = 0 \\ or \\ a_{11}u + a_{12}v + a_{13}c = B_1; \qquad a_{31}u + a_{32}v + a_{33}c = B_3; \qquad \mathbf{n}\cdot\mathbf{V}h = 0 \end{gathered} \tag{3.2.51}$$

3.2.2 Numerical Approximation of Diffusive Wave Equations

Two options are provided in this report to solve the diffusive wave flow equations. One is the finite element method and the other is the particle tracking method.

3.2.2.1 Galerkin Finite Element Method. Recall the diffusive wave is governed by Eq. (2.2.44) which is repeated here as

$$\frac{\partial H}{\partial t} - \nabla\cdot\left[K\left(\nabla H + \frac{h}{2\rho}\nabla(\Delta\rho) - \frac{\tau^s}{\rho g h}\right)\right] = S_S + S_R - S_E + S_I \tag{3.2.52}$$

Applying the Galerkin finite element method to Eq. (3.2.52), we obtain the following matrix equation

$$[M]\frac{d\{H\}}{dt} + [S]\{H\} = \{Q_{\rho w}\} + \{Q_B\} + \{Q_S\} + \{Q_R\} - \{Q_E\} + \{Q_I\} \tag{3.2.53}$$

in which

$$\begin{gathered} M_{ij} = \int_{\Re} N_i N_j d\Re, \qquad S_{ij} = \int_{\Re} \nabla N_i \cdot \mathbf{K} \cdot \nabla N_j d\Re, \\ Q_{\rho wi} = \int_{\Re} \nabla N_i \cdot \mathbf{K} \cdot \left[\frac{h}{2\rho}\nabla(\Delta\rho) - \frac{\boldsymbol{\tau}^s}{hg\rho}\right] d\Re, \;\; Q_{Bi} = \int_B N_i \mathbf{n}\cdot\mathbf{K}\cdot\left[\nabla H + \frac{h}{2\rho}\nabla(\Delta\rho) - \frac{\boldsymbol{\tau}^s}{hg\rho}\right] dB \end{gathered} \tag{3.2.54}$$

$$Q_{Si} = \int_{\Re} N_i S_S d\Re, \quad Q_{Ri} = \int_{\Re} N_i S_R d\Re, \quad Q_{Ei} = \int_{\Re} N_i S_E d\Re, \quad Q_{Ii} = \int_{\Re} N_i S_I d\Re \tag{3.2.55}$$

where N_i and N_j are the base functions of nodes at x_i and x_j, respectively; **n** is the outward-normal unit vector; [M] is the mass matrix, [S] is the stiff matrix, {H} is the solution vector of H, $\{Q_{\rho w}\}$ is the load vector due to density and wind stress effects, $\{Q_B\}$ is the flow rate through the boundary nodes, $\{Q_S\}$ is the flow rate from artificial source/sink, $\{Q_R\}$ is the flow rate from rainfall, $\{Q_E\}$ is the flow rate due to evapotranspiration, and $\{Q_I\}$ is the flow rate to infiltration. It should be noted that $\{Q_I\}$ is the interaction between the overland and subsurface flows.

Approximating the time derivative term in Eq. (3.2.53) with a time-weighted finite difference, we reduce the diffusive equation and its boundary conditions to the following matrix equation

$$[C]\{H\}=\{L\}+\{Q_B\}+\{Q_I\} \qquad \textbf{(3.2.56)}$$

in which

$$[C]=\frac{[M]}{\Delta t}+\theta[S],\quad \{L\}=\left(\frac{[M]}{\Delta t}-(1-\theta)[S]\right)\left\{H^{(n)}\right\}+\left\{Q_{\rho w}\right\}+\{Q_S\}+\{Q_R\}-\{Q_E\} \qquad \textbf{(3.2.57)}$$

where [C] is the coefficient matrix, {L} is the load vector from initial condition, density and wind effects, artificial sink/sources, rainfall, and evapotranspiration; Δt is the time step size; θ is the time weighting factor; and $\{H^{(n)}\}$ is the value of {H} at old time level n. The global boundary conditions must be used to provide $\{Q_B\}$ in Eq. (3.2.56). The interaction between the overland and subsurface flows must be implemented to calculate $\{Q_I\}$. The interactions will be addressed in Section 3.4.

For a global boundary node I, the corresponding algebraic equation from Eq. (3.2.56) is

$$C_{I,1}H_1+\ldots+C_{I,1}H_I+\ldots+C_{I,N}H_N=L_I+Q_{II}+Q_{BI} \qquad \textbf{(3.2.58)}$$

In the above equation there are two unknowns H_I and Q_{BI}; either H_I or Q_{BI}, or the relationship between H_I and Q_{BI} must be specified. The numerical implementation of these boundary conditions is described as follows.

Dirichlet boundary condition: prescribed water depth or stage

If H_I is given on the boundary node I (Dirichlet boundary condition), all coefficients ($C_{I,1}$, ..., $C_{I,I}$, ..., $C_{I,N}$) and right-hand side (L_I and Q_{II}) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then an identity equation is created as

$$H_I=H_{Id},\quad I\in N_D \qquad \textbf{(3.2.59)}$$

where H_{Id} is the prescribed total head on the Dirichlet node I and N_D is the number of Dirichlet boundary nodes. This process is repeated for every Dirichlet nodes. Note it is unnecessary to modify other equations that involving this unknown, which was done in the previous version. By not modifying other equations, the symmetrical property of the matrix is preserved, which makes the iterative solvers more robust. The final set of equations will consist of N_D identity equations and (N - N_D) finite element equations for N unknowns H_i's. After H_i's are obtained, Eq. (3.2.58) is then used to back calculate N_D Q_{BI}'s.

If a direct solver is used to solve the matrix equation, the above procedure will solve N H_i□s accurately except for roundoff errors. However, if an iterative solver is used, a stopping criterion must be strict enough so that the converged solution of N H_i's is accurate enough to the exact solution. With such accurate H_i's, then one can be sure that the back-calculated N_D Q_{BI}'s are accurate.

Flux boundary condition: prescribed flow rate

If Q_{BI} is given (flux boundary condition), all coefficients ($C_{I,1}$, ..., $C_{I,I}$, ..., $C_{I,N}$) and the right-hand side (L_I and Q_{II}) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then Eq. (3.2.58) is modified to incorporate the boundary conditions and used to solve for H_I. The modification of Eq. (3.2.58) is straightforward. Because Q_{BI} is a known quantity, it contributes to the load on the right hand side. This type of boundary conditions is easy to implement. After H_is are obtained, the original Eq. (3.2.58), which is stored in a temporary array, is used to back calculate N_C Q_{BI}'s on flux boundaries (where N_C is the number of flux boundary nodes). These back-calculated Q_{BI}'s should be theoretically identical to the input Q_{BI}'s. However, because of round-off errors (in the case of direct solvers) or because of stopping criteria (in the case of iterative solvers), the back-calculated Q_{BI}'s will be slightly different from the input Q_{BI}'s. If the differences between the two are significant, it is an indication that the solvers have not yielded accurate solutions.

Water depth-dependent boundary condition: prescribed rating curve

If the relationship is given between Q_{BI} and H_I (rating curve boundary condition), all coefficients ($C_{I,1}$, ..., $C_{I,I}$, ..., $C_{I,N}$) and the right-hand side (L_I and Q_{II}) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then Eq. (3.2.58) is modified to incorporate the boundary conditions and used to solve for H_I. The rating-relationship is used to eliminate one of the unknowns, say Q_{BI}, and the modified Eq. (3.2.58) is used to solve for, say H_I. After H_I is solved, the original Eq. (3.2.58) (recall the original Eq. (3.2.58) must be and has been stored in a temporary array) is used to back-calculate Q_{BI}.

3.2.2.2 The Hybrid Lagrangian-Eulerian Finite Element Method. When the hybrid Lagrangian-Eulerian finite element method is used to solve the diffusive wave equation, instead of Eq. (3.2.52), we expand Eq. (2.2.1) to yield following diffusive wave equation in the Lagrangian form

$$\frac{D_V h}{D\tau} + Kh = S_S + S_R - S_E + S_I \quad where \quad K = \nabla \cdot \mathbf{V} \qquad \textbf{(3.2.60)}$$

To use the semi-Lagrangian method to solve the diffusive wave equation, we integrate Eq. (3.2.60) along its characteristic line from x_i at new time level to x_i^* at old time level or on the boundary (Fig. 3.2-3), we obtain

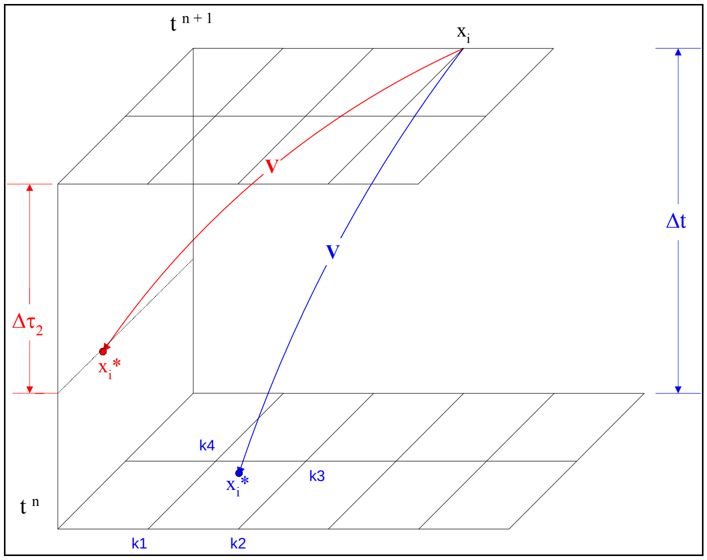

Fig. 3.2-3. Backward Particle Tracking in Two Dimension.

$$\left(1+\frac{\Delta\tau}{2}K_i^{(n+1)}\right)h_i^{(n+1)}=\left(1-\frac{\Delta\tau}{2}K_i^*\right)h_i^*+\frac{\Delta\tau}{2}\left(S_{Si}^{(n+1)}+S_{Si}^*\right)+\frac{\Delta\tau}{2}\left(S_{Ri}^{(n+1)}+S_{Ri}^*\right)-\frac{\Delta\tau}{2}\left(S_{Ei}^{(n+1)}+S_{Ei}^*\right)+\frac{\Delta\tau}{2}\left(S_{Ii}^{(n+1)}+S_{Ii}^*\right) \quad \textbf{(3.2.61)}$$

where $\Delta\tau$ is the tracking time, it is equal to Δt when the backward tracking is carried out all the way to the root of the characteristic and it is less than Δt when the backward tracking hits the boundary before Δt is consumed (Fig. 3.2-3); $K_i^{(n+1)}$), $h_i^{(n+1)}$, $S_{Si}^{(n+1)}$, $S_{Ri}^{(n+1)}$, $S_{Ei}^{(n+1)}$, and $S_{Ii}^{(n+1)}$, respectively, are the values of K, h, S_S, S_R, S_E, and S_I, respectively, at x_i at new time level $t = (n+1)\Delta t$; and K_i^*, h_i^*, S_{Si}^*, S_{Ri}^*, S_{Ei}^*, and S_{Ii}^*, respectively, are the values of K, h, S_S, S_R, S_E, and S_I, respectively, at the location x_i^*. Since the velocity **V** and the decay coefficient K are functions of h, this is a nonlinear hyperbolic problem.

Equation (3.2.61) is solved iteratively to yield the water depth h, and hence the water stage H. The iteration procedure is outlined as follows:

(i) Guess the value of $h^{(k)}$ at the k-th iteration, compute H.
(ii) Apply finite element method to the following equation to obtain V

$$V=\frac{-a}{n}\left[\frac{h}{1+(\nabla Z_O)^2}\right]^{2/3}\frac{1}{\sqrt{\left|-\nabla H-\frac{h}{2\rho}\nabla(\Delta\rho)+\frac{\tau^s}{\rho gh}\right|}}\left(\nabla H+\frac{h}{2\rho}\nabla(\Delta\rho)-\frac{\tau^s}{\rho gh}\right) \quad \textbf{(3.2.62)}$$

(iii) Perform particle tracking to locate x^* and obtain all the *-superscripted quantities.
(iv) Apply the finite element method to the following equation to obtain K

$$K=\nabla\cdot\mathbf{V} \quad \textbf{(3.2.63)}$$

(v) Solve Eq. (3.2.61) along with the boundary condition to obtain new $h^{(k+1)}$

(vi) Check if $h^{(k+1)}$ converges, if yes go to the next time step.

(vii) If $h^{(k+1)}$ does not converge, update h with $h^{(k)} \leftarrow \omega h^{(k+1)} + (1-\omega)h^{(k)}$ and repeat Steps (i) through (vi).

When the wave is transported out of the region at a boundary node (i.e., when $\mathbf{N}\bullet\mathbf{V} \geq 0$), a boundary condition is not needed. When the wave is transported into the region at a node (i.e., when $\mathbf{N}\bullet\mathbf{V} < 0$), a boundary condition must be specified. As in the finite element method, three types of boundary conditions may be encountered.

Dirichlet boundary condition:

For the Dirichlet boundary, the water depth is prescribed as

$$h_1 = h_{Id}, \quad I \in N_D \tag{3.2.64}$$

Flux boundary condition:

For the flux boundary, the flow rate is prescribed as function of time at the boundary node, from which the boundary value is computed as

$$h^{(n+1)} = \frac{q_{up}(t)}{V^{(n+1,k)}} \tag{3.2.65}$$

where $q_{up}(t)$, a function of time t, is the prescribed flow rate [$L^3/t/L$] and $V^{(n+1,k)}$ is the value of V at new time and previous iteration.

Water depth-dependent boundary condition: prescribed rating curve

For the boundary where a rating curve is used to describe the relationship between water depth, h, and volumetric flow rate, q; thus, the water depth, h, on the boundary is computed with

$$V^{(n+1,k)}h^{(n+1)} = f(h) \tag{3.2.66}$$

where f(h) is the rating curve which is a function of h. Equation (3.1.91) is solved iteratively to yield $h^{(n+1)}$.

3.2.3 The Semi-Lagrangian Method for Kinematic Wave

To use the semi-Lagrangian method to solve the kinematic wave equation, Eq. (2.2.50) is rewritten in the Lagrangian form as follows

$$\frac{D_V h}{D\tau} + Kh = S_S + S_R - S_E + S_I \quad where \quad K = \nabla \cdot \mathbf{V} \tag{3.2.67}$$

in which K is the decay coefficient of the wave. Integrating Eq. (3.1.100) along its characteristic line from x_i at new time level to x_i^* at old time level or on the boundary (Fig. 3.2-3), we obtain

$$\left(1+\frac{\Delta\tau}{2}K_i\right)h_i^{(n+1)}=\left(1-\frac{\Delta\tau}{2}K_i^*\right)h_i^*+\frac{\Delta\tau}{2}\left(S_{Si}^{(n+1)}+S_{Si}^*\right)+\frac{\Delta\tau}{2}\left(S_{Ri}^{(n+1)}+S_{Ri}^*\right)$$
$$-\frac{\Delta\tau}{2}\left(S_{Ei}^{(n+1)}+S_{Ei}^*\right)+\frac{\Delta\tau}{2}\left(S_{Ii}^{(n+1)}+S_{Ii}^*\right) \quad \textbf{(3.2.68)}$$

where $\Delta\tau$ is the tracking time, it is equal to Δt when the backward tracking is carried out all the way to the root of the characteristic and it is less than Δt when the backward tracking hits the boundary before Δt is consumed; $K_i^{(n+1)}$), $h_i^{(n+1)}$, $S_{Si}^{(n+1)}$, $S_{Ri}^{(n+1)}$, $S_{Ei}^{(n+1)}$, and $S_{Ii}^{(n+1)}$, respectively, are the values of K, h, S_S, S_R, S_E, and S_I, respectively, at x_i at new time level $t = (n+1)\Delta t$; and K_i^*, h_i^*, S_{Si}^*, S_{Ri}^*, S_{Ei}^*, and S_{Ii}^*, respectively, are the values of K, h, S_S, S_R, S_E, and S_I, respectively, at the location x_i^*. Because of density and wind effects, the velocity **V** and the decay coefficient K are functions of h, this is a nonlinear problem. However, because the nonlinearity due to density and wind effects are normally very weak, Equation (3.2.68) is considered a linear hyperbolic problem with the nonlinear effects evaluated using the values of h at previous time. This equation is used to compute the water depth, h, at all nodes except for the upstream boundary node.

Because the wave is transported into the region at an upstream node, a boundary condition must be specified. The flow rate is normally given as a function of time at an upstream node, from which the boundary value is computed as

$$h_1^{(n+1)}=\frac{q_{up}(t)}{V_i^{(n+1)}} \quad \textbf{(3.2.69)}$$

where $q_{up}(t)$, a function of time t, is the prescribed flow rate [$L^3/t/L$].

3.2.2 Numerical Approximations of Thermal Transport

Two options are provided in this report to solve the thermal transport equation. One is the finite element method and the other is the particle tracking method.

3.2.4.1 Finite Element Method. Recall the thermal transport equation is governed by Eq. (2.2.52) which is rewritten in a slightly different form as

$$\rho_W C_W h\frac{\partial T}{\partial t}+\frac{\partial(\rho_W C_W h)}{\partial t}T+\nabla\cdot\left(\rho_W C_W \mathbf{q}T\right)-\nabla\cdot\left(\mathbf{D}^H h\cdot\nabla T\right)$$
$$=H_a+H_r+H_n-H_b-H_e-H_s+H_i+H_c \quad \textbf{(3.2.70)}$$

Applying the finite element method to Eq. (3.2.70), we obtain the following matrix equation

$$[M]\frac{d\{T\}}{dt}+[V]\{T\}+[D]\{T\}+[K]\{T\}$$
$$=-\left\{\Phi^B\right\}+\left\{\Phi^a\right\}+\left\{\Phi^r\right\}+\left\{\Phi^n\right\}-\left\{\Phi^b\right\}-\left\{\Phi^e\right\}-\left\{\Phi^s\right\}+\left\{\Phi^i\right\}+\left\{\Phi^c\right\} \quad \textbf{(3.2.71)}$$

in which

$$M_{ij} = \int_R N_i \rho_W C_W h N_j dR, \qquad V_{ij} = \int_R \nabla W_i \rho_W C_W \mathbf{q} N_j dR, \qquad D_{ij} = \int_R \nabla N_i \mathbf{D}^{\mathbf{H}} h \nabla N_j dR,$$
$$K_{ij} = \int_R N_i \frac{\partial \rho_W C_W h}{\partial t} N_j dR, \qquad \Phi_i^B = \int_B \mathbf{n} \cdot \left(W_i \rho_W C_W \mathbf{q} T - N_i \mathbf{D}^{\mathbf{H}} h \nabla T \right) dB \tag{3.2.72}$$

$$\Phi_i^a = \int_R N_i H_a R, \qquad \Phi_i^r = \int_R N_i H_r dR, \qquad \Phi_i^n = \int_R N_i H_n dR \tag{3.2.73}$$

$$\Phi_i^b = \int_R N_i H_b dR, \qquad \Phi_i^e = \int_R N_i H_e dR, \qquad \Phi_i^s = \int_R N_i H_s dR, \qquad \Phi_i^c = \int_R N_i H_c dR, \tag{3.2.74}$$

$$\Phi_i^i = \int_R N_i H_i dx \tag{3.2.75}$$

where W_i is the weighting function of node at x_i; N_i and N_j are the base functions of nodes at x_i and x_j, respectively; [M] is the mass matrix, [V] is the stiff matrix due to advective transport; [D] is the stiff matrix due to dispersion/diffusion/conduction; {T} is the solution vector of temperature; $\{\Phi^B\}$ is the vector due to boundary conditions, which can contribute to load vector and/or coefficient matrix; $\{\Phi^a\}$ is the load vector due to artificial energy source; $\{\Phi^r\}$ is the load vector due to energy contained in rainfall; $\{\Phi^n\}$ is the load vector due to net radiation; $\{\Phi^b\}$ is the vector due to backward radiation, which is a nonlinear function of temperature and contributes to both the load vector and coefficient matrix; $\{\Phi^e\}$ is the vector due to energy consumed for evaporation, which is a nonlinear function of temperature and contributes to both the load vector and coefficient matrix; $\{\Phi^s\}$ is the vector due to sensible heat, which is a linear function of temperature and contributes to both the load vector and coefficient matrix; $\{\Phi^c\}$ is the vector due to chemical reaction, which is not considered in this version, but can be added easily; and $\{\Phi^i\}$ is the vector due to interaction with subsurface exfiltraing water.

Approximating the time derivative term in Eq. (3.2.71) with a time-weighted finite difference, we reduce the advective-diffusive equation and its boundary conditions to the following matrix equation

$$[C]\{T\} = \{L\} - \{\Phi^B\} - \{\Phi^b\} - \{\Phi^e\} - \{\Phi^s\} + \{\Phi^i\} \tag{3.2.76}$$

in which

$$[C] = \frac{[M]}{\Delta t} + \theta([D] + [K]) + \theta_V [V],$$
$$\{L\} = \left(\frac{[M]}{\Delta t} - (1-\theta)([DS] + [K]) - (1-\theta_V)[V] \right) \{T^{(n)}\} + \{\Phi^a\} + \{\Phi^r\} + \{\Phi^n\} \tag{3.2.77}$$

where [C] is the coefficient matrix, {L} is the load vector from initial condition, artificial sink/sources, rainfall, and net radiation; Δt is the time step size; θ is the time weighting factor for the dispersion and linear terms; θ_v is the time weighting factor for the velocity term; and $\{T^{(n)}\}$ is the value of {T} at old time level n. The global boundary conditions must be used to provide $\{\Phi^B\}$ in Eq. (3.2.76). The interaction between the overland and subsurface flows must be implemented to calculate $\{\Phi^i\}$. The interactions will be addressed in Section 3.4.

For a global boundary node I, the corresponding algebraic equation from Eq. (3.2.76) is

$$C_{I1}T1 + .. + C_{I,1}TI + .. + C_{I,N}T_N = L_I - \left(\Phi_I^b + \Phi_i^e + \Phi_I^S\right) + \Phi_I^i - \Phi_I^B \qquad \textbf{(3.2.78)}$$

In the above equations there are two unknowns T_I and Φ_I^B; either T_I or Φ_I^B, or the relationship between T_I and Φ_I^B must be specified. The numerical implementation of these boundary conditions is described as follows.

Direchlet boundary condition: prescribed temperature

If T_I is given on the boundary node I (Dirichlet boundary condition), all coefficients ($C_{I,1}$, .., $C_{I,I}$, .., $C_{I,N}$) and the right-hand side terms (L_I, Φ_I^b, Φ_I^e, Φ_I^s, Φ_I^i) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then an identity equation is created as

$$T_I = T_{Id}, \quad I \in N_D \qquad \textbf{(3.2.79)}$$

where T_{Id} is the prescribed temperature on the Dirichlet node I and N_D is the number of Dirichlet boundary nodes. This process is repeated for every Dirichlet nodes. Note it is unnecessary to modify other equations that involving this unknown, which was done in the previous version. By not modifying other equations, the symmetrical property of the matrix is preserved, which makes the iterative solvers more robust. The final set of equations will consist of N_D identity equations and (N - N_D) finite element equations for N unknowns T_i's. After T_i's for all nodes are solved from the matrix equation, Eq. (3.2.78) is then used to back calculate N_D Φ_I^B's.

If a direct solver is used to solve the matrix equation, the above procedure will solve N T_i's accurately except for roundoff errors. However, if an iterative solver is used, a stopping criterion must be strict enough so that the converged solution of N T_i's are accurate enough to the exact solution. With such accurate T_i's, then can be sure that the back-calculated ND Φ_I^B's are accurate.

Cauchy boundary condition: prescribed heat flux

If Φ_{BI} is given (Cauchy flux boundary condition), all coefficients ($C_{I,1}$, .., $C_{I,I}$, .., $C_{I,N}$) and right-hand side terms (L_I, Φ_I^a, Φ_I^r, Φ_I^n, Φ_I^i) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then Eq. (3.2.78) is modified to incorporate the boundary conditions and used to solve for T_I. The modification of Eq. (3.2.78) is straightforward. Because Φ_I^B is a known quantity, it contributes to the load on the right hand side. This type of boundary conditions is very easy to implement. After T_i's are obtained, the original Eq. (3.2.78), which is stored in a temporary array, is used to back calculate N_C Φ_I^B's on flux boundaries (where N_C is the number of flux boundary nodes). These back-calculated Φ_I^B's should be theoretically identical to the input Φ_I^B's. However, because of round-off errors (in the case of direct solvers) or because of stopping criteria (in the case of iterative solvers), the back-calculated Φ_I^B's will be slightly different from the input Φ_I^B's. If the differences between the two are significant, it is an indication that the solvers have not yielded accurate solutions.

Neumann boundary condition: prescribed gradient of temperature

At Neumann boundaries, the temperature gradient is prescribed, thus, the flux due to temperature gradient is given. For this case, all coefficients ($C_{I,1}$, .., $C_{I,I}$, .., $C_{I,N}$) and right-hand side terms (L_I, Φ_I^a, Φ_I^r, Φ_I^n, Φ_I^i) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then Eq. (3.2.78) is modified to incorporate the boundary conditions and used to solve for T_I. For the Neumann boundary condition, Φ_I^B contributes to both the matrix coefficient and load vector, thus both the coefficient matrix [C] and the load vector {L} must be modified. Recall

$$\Phi_i^B = \int_B \mathbf{n} \cdot \left(W_i \rho_W C_W \mathbf{q} T - N_i \mathbf{D}^{\mathbf{H}} h \nabla T \right) dB \tag{3.2.80}$$

Substituting Eq. (2.2.58) into Eq. (3.2.80), we have

$$\begin{gathered} \left\{ \Phi^B \right\} \equiv [CB]\{T\} + \{LB\} \\ \text{in which} \quad CB_{i,j} = \int_B n \cdot W_i \rho_W C_W q N_j dB \quad \text{and} \quad LB_i = \int_B N_i \varphi_{nb}(t) dB \end{gathered} \tag{3.2.81}$$

where [CB] and {LB} are the coefficient matrix and load vector due to Neumann boundary. Adding the I-th equation in Eq. (3.2.81) to Eq. (3.2.78), we obtained a modified equation, which can be solved for solve T_I. After T_I is solved, the original Eq. (3.2.78) (recall the original Eq. (3.2.78) must be and has been stored in a temporary array) is used to back-calculate Φ_I^B.

Variable boundary condition:

At the variable boundary condition Node I, the implementation of boundary conditions can be made identical to that for a Cauchy boundary condition node if the flow is directed into the region. If the flow is going out of the region, the boundary condition is implemented similar to the implementation of Neuman boundary condition with $LB_I = 0$. The assumption of zero Neumann flux implies that a Neuman node must be far away from the source/sink.

3.2.4.2 The Hybrid Lagrangian-Eulerian Finite Element Method. When the hybrid Lagrangian-Eulerian finite element method is used to solve the thermal transport equation, we expand Eq. (3.2.70) to yield following advection-dispersion equation in the Lagrangian form

$$\frac{D_V T}{Dt} + KT = D + \Phi^S + \Phi^I \quad \text{where} \quad \mathbf{V} = \frac{\mathbf{q}}{h} \tag{3.2.82}$$

in which

$$\begin{gathered} K = \frac{1}{\rho_W C_W h} \frac{\partial \rho_W C_W h}{\partial t} + \frac{1}{\rho_W C_W h} \nabla \cdot \left(\rho_W C_W \mathbf{q} \right), \qquad D = \frac{1}{\rho_W C_W h} \nabla \cdot \left(h \mathbf{D}^H \cdot \nabla T \right) \\ \Phi^S = \frac{H_a + H_r + H_n - H_b - H_c - H_s}{\rho_W C_W h}, \qquad \Phi^I = \frac{H_i}{\rho_W C_W h} \end{gathered} \tag{3.2.83}$$

To use the semi-Lagrangian method to solve the thermal transport equation, we integrate Eq. (3.2.82) along its characteristic line from x_i at new time level to x_i^* at old time level or on the boundary (Fig. 3.2-3), we obtain

$$\begin{aligned}\left(1+\frac{\Delta\tau}{2}K_i^{(n+1)}\right)T_1^{(n+1)} &= \left(1-\frac{\Delta\tau}{2}K_i^{*}\right)T_i^{*} + \\ &\frac{\Delta\tau}{2}\left(D_i^{(n+1)}+D_i^{*}\right)+\frac{\Delta\tau}{2}\left(\Phi_i^{S^{(n+1)}}+\Phi_i^{S*}\right)+\frac{\Delta\tau}{2}\left(\Phi_i^{I^{(n+1)}}+\Phi_i^{I^*}\right), \quad i\in N\end{aligned} \qquad \textbf{(3.2.84)}$$

where $\Delta\tau$ is the tracking time, it is equal to Δt when the backward tracking is carried out all the way to the root of the characteristic and it is less than Δt when the backward tracking hits the boundary before Δt is consumed; $K_i^{(n+1)}$, $T_i^{(n+1)}$, $D_i^{(n+1)}$, $\Phi_i^{S(n+1)}$, and $\Phi_i^{I(n+1)}$ respectively, are the values of K, T, D, Φ^S, and Φ^I, respectively, at x_i at new time level $t = (n+1)\Delta t$; and K_i^*, T_i^*, D_i^*, Φ_i^{S*}, and Φ_i^{I*}, respectively, are the values of K, T, D, Φ^S, and Φ^I, respectively, at the location x_i^*.

To compute the dispersion/diffusion terms $D_i^{(n+1)}$ and D_i^*, we rewrite the second equation in Eq. (3.2.83) as

$$\rho_W C_W h D = \nabla\cdot\left(\mathbf{D}^{\mathbf{H}} h\cdot\nabla T\right) \qquad \textbf{(3.2.85)}$$

Applying the Galerkin finite element method to Eq. (3.2.85) at new time level (n+1), we obtain the following matrix equation for $\{D^{(n+1)}\}$ as

$$\left[a^{(n+1)}\right]\left\{D^{(n+1)}\right\}+\left[b^{(n+1)}\right]\left\{T^{(n+1)}\right\}=\left\{B^{(N+1)}\right\} \qquad \textbf{(3.2.86)}$$

in which

$$\left\{D^{(n+1)}\right\}=\left\{D_1^{(n+1)} \quad D_2^{(n+1)} \quad . \; . \quad D_i^{(n+1)} \quad . \; . \quad D_N^{(n+1)}\right\}^{Transpose} \qquad \textbf{(3.2.87)}$$

$$\left\{\mathrm{T}^{(n+1)}\right\}=\left\{\mathrm{T}_1^{(n+1)} \quad \mathrm{T}_2^{(n+1)} \quad . \; . \quad \mathrm{T}_i^{(n+1)} \quad . \; . \quad \mathrm{T}_N^{(n+1)}\right\}^{\mathrm{Transpose}} \qquad \textbf{(3.2.88)}$$

$$\left\{\mathbf{B}^{(n+1)}\right\}=\left\{B_1^{(n+1)} \quad B_2^{(n+1)} \quad . \; . \quad B_i^{(n+1)} \quad . \; . \quad B_N^{(n+1)}\right\}^{Transpose} \qquad \textbf{(3.2.89)}$$

$$\begin{aligned}a_{ij}^{(n+1)} &= \int_R N_i\left(\rho_W C_W h\right)\Big|_{(n+1)} N_j dR, \qquad b_{ij}^{(n+1)} = \int_R \nabla N_i\cdot\left(\mathbf{D}^{\mathbf{H}} h\right)\Big|_{(n+1)}\cdot\nabla N_j dR, \\ B_i^{(n+1)} &= \int_B \mathbf{n}\cdot N_i\left(\mathbf{D}^{\mathbf{H}} h\right)\Big|_{(n+1)}\cdot\nabla T^{(n+1)} dB\end{aligned} \qquad \textbf{(3.2.90)}$$

where the superscript (n+1) denotes the time level; N and N are the base functions of nodes at x_i and x_j, respectively.

Lumping the matrix $[a^{(n+1)}]$, we can solve Eq. (3.2.86) for $D_I^{(n+1)}$ as follows

$$\begin{aligned}D_I^{(n+1)} &= -\frac{1}{a_{II}^{(n+1)}}\sum_j b_{Ij}^{(n+1)} T_j^{(n+1)} \quad if \quad I \quad is\ an\ interior\ point \\ D_I^{(n+1)} &= \frac{1}{a_{II}^{(n+1)}} B_I^{(n+1)} - \frac{1}{a_{II}^{(n+1)}}\sum_j b_{Ij}^{(n+1)} T_j^{(n+1)} \quad if \quad I \quad is\ a\ boundary\ point\end{aligned} \qquad \textbf{(3.2.91)}$$

where $a_{II}^{(n+1)}$ is the lumped $a_{ii}^{(n+1)}$. Following the identical procedure that leads Eq. (3.2.85) to Eq. (3.2.91), we have

$$
\begin{aligned}
D_I^{(n)} &= -\frac{1}{a_{II}^{(n)}}\sum_j b_{Ij}^{(n)} T_j^{(n)} \quad \textit{if I is an interior point} \\
D_I^{(n)} &= \frac{1}{a_{II}^{(n)}} B_I^{(n)} - \frac{1}{a_{II}^{(n)}}\sum_j b_{Ij}^{(n)} T_j^{(n)} \quad \textit{if I is a boundary point}
\end{aligned}
\qquad \textbf{(3.2.92)}
$$

where $\{B^{(n)}\}$, $\{a^{(n)}\}$ and $\{b^{(n)}\}$, respectively, are defined similar to $\{B^{(n+1)}\}$, $\{a^{(n+1)}\}$ and $\{b^{(n+1)}\}$, respectively.

With $\{D^{(n)}\}$ calculated with Eq. (3.2.92), $\{D^*\}$ can be interpolated. Substituting Eq. (3.2.91) into Eq. (3.2.84) and implementing boundary conditions given in Section 2.2.4, we obtain a system of N simultaneous algebraic equations N unknowns ($T_i^{(n+1)}$ for i = 1, 2, .., N.) If the dispersion/diffusion term is not included, then Eq. (3.2.84) is reduced to a set of N decoupled equations as

$$a_{ii} T_i^{(n+1)} = b_i, \quad i \in N \qquad \textbf{(3.2.93)}$$

where

$$a_{ii} = \left(1 + \frac{\Delta\tau}{2} K_i^{(n+1)}\right) \qquad \textbf{(3.2.94)}$$

$$b_i = \left(1 - \frac{\Delta\tau}{2} K_i^*\right) T_i^* + \frac{\Delta\tau}{2}\left(\Phi_i^{S^{(n+1)}} + \Phi i_i^{S*}\right) + \frac{\Delta\tau}{2}\left(\Phi_i^{I^{(n+1)}} + \Phi_i^{I*}\right), \quad i \in N \qquad \textbf{(3.2.95)}$$

Equation (3.2.93) is applied to all interior nodes without having to make any modification. On a boundary point, there two possibilities: Eq. (3.2.93) is replaced with a boundary equation when the flow is directed into the region or Eq. (3.2.93) is still valid when the flow is direct out of the region. In other words, when the thermal energy is transported out of the region at a boundary node (i.e., when **N•V** ≥ 0), a boundary condition is not needed and Equation (3.2.93) is used to compute the $T_i^{(n+1)}$. When the thermal energy is transported into the region at a node (i.e., when **N•V** < 0), a boundary condition must be specified.

Alternatively, to facilitate the implementation of boundary condition at incoming flow node, the algebraic equation for the boundary node is obtained by applying the finite element method to the boundary node. For this alternative approach, the implementation of boundary conditions at global boundary nodes is identical to that in the finite element approximation of solving the thermal transport equation.

3.2.4 Numerical Approximations of Salinity Transport

Two options are provided in this report to solve the salinity transport equation. One is the finite element method and the other is the particle tracking method.

3.2.5.1 Finite Element Method. Recall the salinity transport equation is governed by Eq. (2.2.60) which is rewritten in a slightly different form as

$$h\frac{\partial S}{\partial t}+\frac{\partial h}{\partial t}S+\nabla\cdot(\mathbf{q}S)-\nabla\cdot\left(\mathbf{D}^{S}h\cdot\nabla h\right)=M_s^{as}+M_x^{rs}-M_s^{es}+M_s^{is} \qquad \textbf{(3.2.96)}$$

Applying the finite element method to Eq. (3.2.96), we obtain the following matrix equation

$$[M]\frac{d\{S\}}{dt}+[V]\{S\}+[D]\{S\}+[K]\{S\}=-\{\Psi^B\}+\{\Psi^a\}+\{\Psi^r\}-\{\Psi^e\}+\{\Psi^i\} \qquad \textbf{(3.2.97)}$$

in which

$$M_{ij}=\int_R N_i h N_j dx, \qquad V_{ij}=\int_R \nabla W_i\cdot\mathbf{q}N_j dR, \qquad D_{ij}=\int_R \nabla N_i\cdot\mathbf{D}^{S}h\cdot\nabla N_j dR,$$
$$K_{ij}=\int_r N_i\frac{\partial h}{\partial t}N_j dR, \qquad \Psi_i^B=\int_B \mathbf{n}\cdot\left(W_i\mathbf{q}S-N_i\mathbf{D}^{S}h\cdot\nabla S\right)dB \qquad \textbf{(3.2.98)}$$

$$\Psi_i^a=\int_R N_i M_S^{as}dR,\ \ \Psi_i^r=\int_R N_i M_s^{rs}dR,\ \ \Psi_i^e=\int_R N_i M_s^{es}dR,\ \ \Psi_i^i=\int_R N_i M_s^{is}dR \qquad \textbf{(3.2.99)}$$

where W_i is the weighting function of node at x_i; N_i and N_j are the base functions of nodes at x_i and x_j, respectively; [M] is the mass matrix, [V] is the stiff matrix due to advective transport; [D] is the stiff matrix due to dispersion/diffusion/conduction; [K] is the stiff matrix due to the linear term; {S} is the solution vector of salinity; $\{\Psi^B\}$ is the vector due to boundary conditions, which can contribute to load vector and/or coefficient matrix; $\{\Psi^a\}$ is the load vector due to artificial salt source; $\{\Psi^r\}$ is the load vector due to salt in rainfall; $\{\Psi^e\}$ is the vector due to evapotranspiration, which is most likely to be zero; and $\{\Psi^i\}$ is the vector due to interaction with subsurface exfiltraing water.

Approximating the time derivative term in Eq. (3.2.97) with a time-weighted finite difference, we reduce the advective-diffusive equation and its boundary conditions to the following matrix equation.

$$[C]\{S\}=\{L\}-\{\Psi^B\}+\{\Psi^i\} \qquad \textbf{(3.2.100)}$$

in which

$$[C]=\frac{[M]}{\Delta t}+\theta([D]+[K])+\theta_V[V],$$
$$\{L\}=\left(\frac{[M]}{\Delta t}-(1-\theta)([D]+[K])-(1-\theta_V)[V]\right)\{S^{(n)}\}+\{\Psi^a\}+\{\Psi^r\} \qquad \textbf{(3.2.101)}$$

where [C] is the coefficient matrix, {L} is the load vector from initial condition, artificial sink/sources and rainfall; Δt is the time step size; θ is the time weighting factor for the dispersion and linear terms; θ_v is the time weighting factor for the velocity term; and $\{S^{(n)}\}$ is the value of {S} at old time level n. The global boundary conditions must be used to provide $\{\Psi^B\}$ in Eq. (3.2.100). The interaction between the overland and subsurface flows must be implemented to calculate $\{\Psi^i\}$. The interactions will be addressed in Section 3.4.

For a global boundary node I, the corresponding algebraic equation from Eq. (3.2.100) is

$$C_{I,1}S_1 + .. + C_{I,I}S_I + .. + C_{I,N}S_N = L_I + \Psi_I^i - \Psi_I^B \qquad \textbf{(3.2.102)}$$

In the above equations there are two unknowns T_I and Ψ_I^B; either T_I or Ψ_I^B, or the relationship between T_I and Ψ_I^B must be specified. The numerical implementations of these boundary conditions are described as follows.

Dirichlet boundary condition: prescribed salinity

If S_I is given on the boundary node I (Dirichlet boundary condition), all coefficients ($C_{I,1}$, .., $C_{I,I}$, .., $C_{I,N}$) and the right-hand side terms (L_I and Ψ_I^i) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then an identity equation is created as

$$S_I = S_{Id}, \quad I \in N_D \qquad \textbf{(3.2.103)}$$

where S_{Id} is the prescribed salinity on the Dirichlet node I and N_D is the number of Dirichlet boundary nodes. This process is repeated for every Dirichlet nodes. Note it is unnecessary to modify other equations that involving this unknown, which was done in the previous version. By not modifying other equations, the symmetrical property of the matrix is preserved, which makes the iterative solvers more robust. The final set of equations will consist of N_D identity equations and (N - N_D) finite element equations for N unknowns S_i's. After S_i's for all nodes are solved from the matrix equation, Eq. (3.2.100) is then used to back calculate N_D Ψ_I^B's.

If a direct solver is used to solve the matrix equation, the above procedure will solve N S_i's accurately except for roundoff errors. However, if an iterative solver is used, a stopping criterion must be strict enough so that the converged solution of N S_i's are accurate enough to the exact solution. With such accurate S_i's, then can be sure that the back-calculated ND Ψ_{BI}'s are accurate.

Cauchy boundary condition: prescribed salt flux

If Ψ_I^B is given (Cauchy flux boundary condition), all coefficients ($C_{I,1}$, .., $C_{I,I}$, .., $C_{I,N}$) and the right-hand side terms (L_I and Ψ_I^i) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then Eq. (3.2.102) is modified to incorporate the boundary conditions and used to solve for S_I. The modification of Eq. (3.2.102) is straightforward. Because Ψ_I^B is a known quantity, it contributes to the load on the right hand side. This type of boundary conditions is very easy to implement. After S_i's are obtained, the original Eq. (3.2.102), which is stored in a temporary array, is used to back calculate N_C Ψ_I^B's on flux boundaries (where N_C is the number of flux boundary nodes). These back-calculated Ψ_I^B's should be theoretically identical to the input Ψ_I^B's. However, because of round-off errors (in the case of direct solvers) or because of stopping criteria (in the case of iterative solvers), the back-calculated Ψ_I^B's will be slightly different from the input Ψ_I^B's. If the differences between the two are significant, it is an indication that the solvers have not yielded accurate solutions.

Neumann boundary condition: prescribed gradient of salinity

At Neumann boundaries, the temperature gradient is prescribed, thus, the flux due to temperature gradient is given. For this case, all coefficients ($C_{I,}$, .., $C_{I,I}$, .., $C_{I,N}$) and the right-hand side terms (L_I and Ψ_I^i) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then Eq. (3.2.102) is modified to incorporate the boundary conditions and used to solve for S_I. For the Neumann boundary condition, Ψ_I^B contributes to both the matrix coefficient and load vector, thus both the coefficient matrix [C] and the load vector {L} must be modified. Recall

$$\Psi_i^B = \int_B \mathbf{n}\cdot\left(W_i\mathbf{q}S - N_i D^S h\nabla S\right)dB \tag{3.2.104}$$

Substituting Eq. (2.2.66) into Eq. (3.2.104), we have

$$\begin{aligned}&\left\{\Psi^B\right\} \equiv \left[CB\right]\left\{S\right\} + \left\{LB\right\}\\ &\textit{in which} \qquad CB_{i,j} = \int_B \mathbf{n}\cdot W_i\mathbf{q}N_j dB \quad \textit{and} \quad LB_i = \int_B N_i\Psi_{nb}(t)dB\end{aligned} \tag{3.2.105}$$

where [CB] and {LB} are the coefficient matrix and load vector due to Neumann boundary. Adding the I-th equation in Eq. (3.2.105) to Eq. (3.2.102), we obtained a modified equation, which can be solved for solve S_I. After S_I is solved, the original Eq. (3.2.102) (recall the original Eq. (3.2.102) must be and has been stored in a temporary array) is used to back-calculate Ψ_I^B.

Variable boundary condition:

At the variable boundary condition Node I, the implementation of boundary conditions can be made identical to that for a Cauchy boundary condition node if the flow is directed into the river/stream/canal reach. If the flow is going out of the reach, the boundary condition is implemented similar to the implementation of Neuman boundary condition with $\Psi_I^{nb} = 0$. The assumption of zero Neumann flux implies that a Neuman node must be far away from the source/sink.

3.2.5.2 The Hybrid Lagrangian-Eulerian Finite Element Method. When the hybrid Lagrangian-Eulerian finite element method is used to solve the salt transport equation, we expand Eq. (3.2.96) to yield following advection-dispersion equation in the Lagrangian form

$$\frac{D_V S}{Dt} + KS = D + \Psi^S + \Psi^I \quad \textit{where} \quad \mathbf{V} = \frac{\mathbf{q}}{h} \tag{3.2.106}$$

in which

$$K = \frac{1}{h}\frac{\partial h}{\partial t} + \frac{1}{h}\nabla\cdot(\mathbf{q}), \qquad D = \frac{1}{h}\nabla\cdot\left(h\mathbf{D}^S\cdot\nabla S\right), \qquad \Psi^S = \frac{M_s^{as} + M_s^{rs} - M_s^{es}}{h}, \qquad \Psi^I = \frac{M_s^{is}}{h} \tag{3.2.107}$$

To use the semi-Lagrangian method to solve the thermal transport equation, we integrate Eq. (3.2.106) along its characteristic line from x_i at new time level to x_i^* at old time level or on the boundary (Fig. 3.2-3), we obtain

$$\left(1+\frac{\Delta\tau}{2}K_i^{(n+1)}\right)S_i^{(n+1)}=\left(1-\frac{\Delta\tau}{2}K_i^*\right)S_i^*+\frac{\Delta\tau}{2}\left(D_i^{(n+1)}+D_i^*\right)$$
$$+\frac{\Delta\tau}{2}\left(\Psi_i^{S(n+1)}+\Psi_i^{S*}\right)+\frac{\Delta\tau}{2}\left(\Psi_i^{I(n+1)}+\Psi_i^{I*}\right),\quad i\in N \qquad \textbf{(3.2.108)}$$

where $\Delta\tau$ is the tracking time, it is equal to Δt when the backward tracking is carried out all the way to the root of the characteristic and it is less than Δt when the backward tracking hits the boundary before Δt is consumed; $K_i^{(n+1)}$, $T_i^{(n+1)}$, $D_i^{(n+1)}$, $\Psi_i^{S(n+1)}$, and $\Psi_i^{I(n+1)}$ respectively, are the values of K, T, D, Ψ^S, and Ψ^I, respectively, at x_i at new time level t = (n+1)Δt; and K_i^*, T_i^*, D_i^*, Ψ_i^{S*}, and Ψ_i^{I*}, respectively, are the values of K, T, D, Ψ^S, and Ψ^I, respectively, at the location x_i^*.

To compute the dispersion/diffusion terms $D_i^{(n+1)}$ and D_i^*, we rewrite the second equation in Eq. (3.2.107) as

$$hD=\nabla\cdot\left(h\mathbf{D}^S\cdot\nabla S\right) \qquad \textbf{(3.2.109)}$$

Applying the Galerkin finite element method to Eq. (3.2.109) at new time level (n+1), we obtain the following matrix equation for $\{D^{(n+1)}\}$ as

$$\left[a^{(n+1)}\right]\left\{D^{(n+1)}\right\}+\left[b^{(n+1)}\right]\left\{S^{(n+1)}\right\}=\left\{B^{(n+1)}\right\} \qquad \textbf{(3.2.110)}$$

in which

$$\left\{D^{(n+1)}\right\}=\left\{D_1^{(n+1)}\quad D_2^{(n+1)}\quad .\ .\quad D_i^{(n+1)}\quad .\ .\quad D_N^{(n+1)}\right\}^{Transpose} \qquad \textbf{(3.2.111)}$$

$$\left\{S^{(n+1)}\right\}=\left\{S_1^{(n+1)}\quad S_2^{(n+1)}\quad .\ .\quad S_i^{(n+1)}\quad .\ .\quad S_N^{(n+1)}\right\}^{Transpose} \qquad \textbf{(3.2.112)}$$

$$\left\{B^{(n+1)}\right\}=\left\{B_1^{(n+1)}\quad B_2^{(n+1)}\quad .\ .\quad B_i^{(n+1)}\quad .\ .\quad B_N^{(n+1)}\right\}^{Transpose} \qquad \textbf{(3.2.113)}$$

$$a_{ij}^{(n+1)}=\int_R N_i\left(h\right)\Big|_{(n+1)}N_j\,dR,\qquad b_{ij}^{(n+1)}=\int_R\nabla N_i\cdot\left(h\mathbf{D}^S\right)\Big|_{(n+1)}\cdot\nabla N_j\,dR,$$
$$B_i^{(n+1)}=\int_B\mathbf{n}\cdot N_i\left(h\mathbf{D}^S\right)\Big|_{(n+1)}\cdot\nabla S^{(n+1)}dB \qquad \textbf{(3.2.114)}$$

where the superscript (n+1) denotes the time level; N and N are the base functions of nodes at x_i and x_j, respectively.

Lumping the matrix $[a^{(n+1)}]$, we can solve Eq. (3.2.110) for $D_I^{(n+1)}$ as follows

$$D_I^{(n+1)}=-\frac{1}{a_{II}^{(n+1)}}\sum_j b_{Ij}^{(n+1)}S_j^{(n+1)}\quad if\quad I\ is\ an\ interior\ point$$
$$D_I^{(n+1)}=\frac{1}{a_{II}^{(n+1)}}B_I^{(n+1)}-\frac{1}{a_{II}^{(n+1)}}\sum_j b_{Ij}^{(n+1)}S_j^{(n+1)}\quad if\quad I\ is\ a\ boundary\ point \qquad \textbf{(3.2.115}$$

where $a_{II}^{(n+1)}$ is the lumped $a_{ii}^{(n+1)}$. Following the identical procedure that leads Eq. (3.2.109) to Eq.

(3.2.115), we have

$$D_I^{(n)} = -\frac{1}{a_{II}^{(n)}}\sum_j b_{Ij}^{(n)} S_j^{(n)} \quad if \quad I \text{ is an interior point}$$
$$D_I^{(n)} = \frac{1}{a_{II}^{(n)}} B_I^{(n)} - \frac{1}{a_{II}^{(n)}}\sum_j b_{Ij}^{(n)} S_j^{(n)} \quad if \quad I \text{ is a boundary point} \qquad \textbf{(3.2.116)}$$

where $\{B^{(n)}\}$, $\{a^{(n)}\}$ and $\{b^{(n)}\}$, respectively, are defined similar to $\{B^{(n+1)}\}$, $\{a^{(n+1)}\}$ and $\{b^{(n+1)}\}$, respectively.

With $\{D^{(n)}\}$ calculated with Eq. (3.2.116), $\{D^*\}$ can be interpolated. Substituting Eq. (3.2.115) into Eq. (3.2.108) and implementing boundary conditions given in Section 2.2.5, we obtain a system of N simultaneous algebraic equations N unknowns ($S_i^{(n+1)}$ for i = 1, 2, .., N.) If the dispersion/diffusion term is not included, then Eq. (3.2.108) is reduced to a set of N decoupled equations as

$$a_{ii} S_i^{(n+1)} = b_i, \quad i \in N \qquad \textbf{(3.2.117)}$$

where

$$a_{ii} = \left(1 + \frac{\Delta\tau}{2} K_i^{(n+1)}\right) \qquad \textbf{(3.2.118)}$$

$$b_i = \left(1 - \frac{\Delta\tau}{2} K_i^*\right) S_i^* + \frac{\Delta\tau}{2}\left(\Psi_i^{S^{(n+1)}} + \Psi_i^{S^*}\right) + \frac{\Delta\tau}{2}\left(\Psi_i^{I^{(n+1)}} + \Psi_i^{I^*}\right), \quad i \in N \qquad \textbf{(3.2.119)}$$

Equation (3.2.117) is applied to all interior nodes without having to make any modification. On a boundary point, there two possibilities: Eq. (3.2.117) is replaced with a boundary equation when the flow is directed into the region or Eq. (3.2.117) is still valid when the flow is direct out of the region. In other words, when the salt is transported out of the region at a boundary node (i.e., when $\mathbf{N}\bullet\mathbf{V} \geq 0$), a boundary condition is not needed and Equation (3.2.117) is used to compute the $S_i^{(n+1)}$. When the salt is transported into the region at a node (i.e., when $\mathbf{N}\bullet\mathbf{V} < 0$), a boundary condition must be specified.

Alternatively, to facilitate the implementation of boundary condition at incoming flow node, the algebraic equation for the boundary node is obtained by applying the finite element method to the boundary node. For this alternative approach, the implementation of boundary conditions at global boundary nodes is identical to that in the finite element approximation of solving the salt transport equation.

3.3 Solving the Three-Dimensional Subsurface Flow Equations

The Richards equation is discretized with the Galerkin finite element method in space and with the finite difference method in time. In our model, the steady-state version of subsurface flow equations can be solved for determining the initial subsurface flow condition when boundary conditions are complicated and/or unsaturated zones are taken into account. The details of solving the Richards equation and the salt transport has been described in detail elsewhere (Yeh et al, 1994; Lin et al., 1997). The numerical solution of thermal transport equations follows similar to that for two-dimensional thermal equation in overland flow. These numerical solutions are summarized below for the completeness of this report.

3.3.1 Finite Element Approximations of the Flow Equations

Finite element disretization in space. When using the finite element method, the referenced pressure head in Eq. (2.3.1) is approximated by:

$$h \approx \hat{h} = \sum_{j=1}^{N} h_j(t) N_j(x,y,z) \qquad \textbf{(3.3.1)}$$

where h_j and N_j are the amplitude of h and the base function, respectively, at nodal point j and N is the total number of nodes. After defining a residual and forcing the weighted residual to zero, the flow equation, Eq.(2.3.1), is approximated as:

$$\begin{aligned} &\left[\int_R N_i \frac{\rho}{\rho_o} F N_j dR\right]\frac{dh_j}{dt} + \left[\int_R (\nabla N_i)\cdot \mathbf{K}\cdot(\nabla N_j) dR\right] h_j \\ &= \int_R N_i \frac{\rho^*}{\rho_o} q\, dR - \int_R (\nabla N_i)\cdot\mathbf{K}\cdot\frac{\rho}{\rho_o}\nabla z dR + \int_B \mathbf{n}\cdot\mathbf{K}\cdot\left(\nabla h + \frac{\rho}{\rho_o}\nabla z\right) N_i dB \end{aligned} \qquad \textbf{(3.3.2)}$$

In matrix form, Eq.(3.3.2) is written as:

$$[M]\left\{\frac{dh}{dt}\right\} + [S]\{h\} = \{Q\} + \{G\} + \{B\} \qquad \textbf{(3.3.3)}$$

where {dh/dt} and {h} are the column vectors containing the values of dh/dt and h, respectively, at all nodes; [M] is the mass matrix resulting from the storage term; [S] is the stiff matrix resulting from the action of conductivity; {Q}, {G}, and {B} are the load vectors from the internal source/sink, gravity force, and boundary conditions, respectively. The mass matrix, [M], and stiff matrix, [S], are defined as:

$$M_{ij} = \sum_{e \in M_e} \int_{R_e} N_\alpha^e \frac{\rho}{\rho_o} F N_\beta^e dR \quad and \quad S_{ij} = \sum_{e \in M_e} \int_{R_e} (\nabla N_\alpha^e)\cdot\mathbf{K}\cdot(\nabla N_\beta^e) dR \qquad \textbf{(3.3.4)}$$

where R_e is the region of element e, M_e is the set of elements that have a local side α-β coinciding with the global side i-j, and N_α^e is the α-th local base function of element e. The three load vectors, {Q}, {G}, and {B}, are defined as:

$$Q_i = \sum_{e \in M_e} \int_{R_e} N_\alpha^e \frac{\rho}{\rho_o} q dR, \qquad G_i = -\sum_{e \in M_e} \int_{R_e} (\nabla N_\alpha^e)\cdot\mathbf{K}\cdot\frac{\rho}{\rho_o}\nabla z\, dR \qquad \textbf{(3.3.5)}$$

$$B_i = -\sum_{e \in N_{se}} \int_{B_e} N_\alpha^e n \cdot \left[-K \cdot \left\{ \nabla h + \frac{\rho}{\rho_o} \nabla z \right\} \right] dB \qquad \textbf{(3.3.6)}$$

where N_{se} is the set of boundary segments that have a local node α coinciding with the global node i, and B_e is the length of boundary segment e.

Finite element evaluation of Darcy velocity. In most numerical models, Darcy velocity components are calculated numerically by taking the derivatives of the simulated h as

$$\mathbf{V} = -\mathbf{K} \cdot \left(\frac{\rho}{\rho_o} (\nabla N_j) h_j + \nabla z \right) \qquad \textbf{(3.3.7)}$$

The above formulation results in velocity field which is not continuous at element boundaries and nodal points if the variation of h is other than linear or constants. The alternative approach would be to apply the Galerkin finite element method to Eq. (2.3.3), thus one obtains

$$[U]\{V_x\} = \{D_x\}; \quad [U]\{V_y\} = \{D_y\}; \quad [U]\{V_z\} = \{D_z\} \qquad \textbf{(3.3.8)}$$

where the matrix [U] and the load vectors $\{D_x\}$, $\{D_y\}$, and $\{D_z\}$ are given by

$$U_{ij} = \sum_{e \in M_e} \int_{R_e} N_\alpha^e N_\beta^e dR, \qquad D_{xi} = \sum_{e \in M_e} \int_{R_e} N_\alpha^e \mathbf{i} \cdot \mathbf{K} \cdot \left\{ \frac{\rho_o}{\rho} \nabla h + \nabla z \right\} dR, \qquad \textbf{(3.3.9)}$$

$$D_{yi} = -\sum_{e \in M_e} \int_{R_e} N_\alpha^e \mathbf{j} \cdot \mathbf{K} \cdot \left\{ \frac{\rho_o}{\rho} \nabla h + \nabla z \right\} dR, \qquad D_{zi} = \sum_{e \in M_e} \int_{R_e} N_\alpha^e \mathbf{k} \cdot \mathbf{K} \cdot \left\{ \frac{\rho_o}{\rho} \nabla h + \nabla z \right\} dR \qquad \textbf{(3.3.10)}$$

where V_x, V_y, and V_z are the Darcy velocity components along the x-, y-, and z-directions, respectively and **i**, **j**, and **k** are the unit vector along the x-, y-, and z-coordinates, respectively.

Finite difference discretization in time. We derive a matrix equation by integrating Eq. (3.3.3). An important advantage in finite element approximation over the finite difference approximation is the inherent ability to handle complex boundaries and obtain the normal derivatives therein. In the time dimension, such advantages are not evident. Thus, finite difference methods are typically used in the approximation of the time derivative. Two time-marching methods are adopted in the present model.

The first one is the time weighted method written as:

$$\frac{[M]}{\Delta t}\left(\{h\}_{t+\Delta t} - \{h\}_t \right) + \omega [S] \{h\}_{t+\Delta t} + (1-\omega)[S]\{h_t\} = \{Q\} + \{G\} + \{B\} \qquad \textbf{(3.3.11)}$$

where [M], [S], {Q}, {G}, and {B} are evaluated at (t + ωΔt). In the Crank-Nicolson centered-in-time approach ω = 0.5, in the backward-difference (implicit difference) ω = 1.0, and in the forward-difference (explicit scheme) ω = 0.0. The central-Nicolson algorithm has a truncation error of $O(\Delta t^2)$, but its propagation-of-error characteristics frequently lead to oscillatory nonlinear instability.

Both the backward-difference and forward-difference have a truncation error of O(Δt). The backward-difference is quite resistant to oscillatory nonlinear instability. On the other hand, the forward difference is only conditionally stable even for linear problems, not to mention nonlinear problems.

In the second method, the values of unknown variables are assumed to vary linearly with time during the time interval, Δt. In this mid-difference method, the recurrence formula is written as:

$$\left(\frac{2}{\Delta t}[M]+[S]\right)\{h\}_{t+\Delta t/2}-\frac{2}{\Delta t}[M]\{h\}_t=\{Q\}+\{G\}+\{B\} \quad \textbf{(3.3.12)}$$

and

$$\{h\}_{t+\Delta t}=2\{h\}_{t+\Delta t/2}-\{h\}_t, \quad \textbf{(3.3.13)}$$

where [M], [S], {Q}, and {B} are evaluated at (t+Δt/2).

Equations (3.3.11) and (3.3.12) can be written as a matrix equation

$$[A]\{h\}=\{L\}+\{B\}, \quad \textbf{(3.3.14)}$$

where [A] is the assembled coefficient matrix, {h} is the unknown vector to be found and represents the values of discretized pressure field at new time, {L} is the load vector due to initial conditions and all types of sources/sinks, and {B} is the load vector due to boundary conditions including the global boundary and media-interface boundaries. Take for example, Eq. (3.3.11) with ω = 1.0, [C] and {L} represent the following:

$$[A]=\frac{[M]}{\Delta t}+[S] \quad and \quad \{L\}=\frac{[M]}{\Delta t}\{h\}_t+\{Q\}+\{G\} \quad \textbf{(3.3.15)}$$

where $\{h\}_t$ is the vector of the discretized pressure field at previous time.

Mass lumping. Referring to the mass matrix, [M], one may recall that this is a unit matrix if the finite difference formulation is used in spatial discretization. Hence, by proper scaling, the mass matrix can be reduced to the finite-difference equivalent by lumping (Clough 1971). In many cases, the lumped mass matrix would result in better solution, in particular, if it is used in conjunction with the central or backward-difference time marching (Yeh and Ward 1980). Under such circumstances, it is preferred to the consistent mass matrix (mass matrix without lumping). Therefore, options are provided for the lumping of the matrix [M]. More explicitly, [M] will be lumped according to:

$$M_{ij}=\sum_{e\varepsilon Me}\left(\sum_{\beta=1}^{N_e}\int_{R_e}N_\alpha^e\frac{\rho}{\rho_o}FN_\beta^e dR\right) \quad if \quad j=i \quad and \quad M_{ij}=0 \quad if \quad j\neq i \quad \textbf{(3.3.16)}$$

Implementation of global Boundary Conditions. For any interior node I, its algebraic equation is obtained by the I-th row of Eq. (3.3.14) as

$$A_{I,1}h_1+\ldots+A_{I,I}h_I+\ldots+A_{I,N}h_N=L_I \quad \textbf{(3.3.17)}$$

Note that B_I is absent from Eq. (3.3.17) for all interior nodes. For the purpose of discussion, one may consider Eq. (3.3.17) to correspond the unknown h_I (one equation, one unknown). For any boundary node I, the corresponding algebraic equation from Eq. (3.3.14) is

$$A_{I,1}h_1 + \ldots + A_{I,I}h_I + \ldots + A_{I,N}h_N = L_I + B_I \qquad \textbf{(3.3.18)}$$

In the above equation there are two unknowns h_I and B_I; either h_I or B_I, or the relationship between h_I and B_I must be specified. Before the implementation of global boundary and media-interface boundary conditions, the coefficient matrix ($A_{I,1}$, .., $A_{I,I}$, .., $A_{I,N}$) and the right hand load term (L_I) must be stored in a temporary array. Then Eq. (3.3.18) is modified with the implementation of boundary conditions. After the implementation, the modified equations are solved for the primary unknown h_I's. The final step is to back calculate B_I's using unmodified Eq. (3.3.18).

The global and interface (river-subsurface media interface or overland-subsurface media interface) conditions must be used to provide {B} for all boundary nodes in Eq. (3.3.18). The interface boundary condition will be addressed in Sub-sections 3.4.2 through 3.4.4. The global boundary conditions are addressed below.

Dirichlet boundary condition: prescribed pressure head

For a Dirichlet node I, we simply rewrite Eq. (3.3.18) as

$$h_I = h_d \qquad \textbf{(3.3.19)}$$

which is obtained by modifying both the corresponding coefficient matrix and load vector as

$$A_{I,1} = 0, .., A_{I,I-1} = 0, A_{I,I+1} = 1, A_{I,I+1} = 0, .., A_{I,N} = 0 \quad and \quad L_I + B_I = h_d \qquad \textbf{(3.3.20)}$$

Thus, it is seen that for a Dirichlet node, both the matrix coefficient and the load vector are modified.

Cauchy boundary condition: prescribed total flux

For the Cauchy boundary condition given by Eq.(2.3.7), we simply substitute Eq.(2.3.7) into Eq.(3.3.6) to yield the value of B_I for the Cauchy node I:

$$B_I = -\int_{B_c} N_I \frac{\rho}{\rho_o} q_c dB \,, \qquad \textbf{(3.3.21)}$$

Thus, the modification of Eq. (3.3.18) is to simply add B_I to L_I.

Neumann boundary condition: prescribed gradient flux

For the Neumann boundary condition given by Eq.(2.3.6), we substitute Eq.(2.3.6) into Eq.(3.3.6) to yield the value of B_I for the Neumann node I:

$$B_I = \int N_I \left(\mathbf{n} \cdot \mathbf{K} \cdot \frac{\rho}{\rho_o} \nabla z - q_n \right) dB \tag{3.3.22}$$

If the hydraulic conductivity is evaluated using the value of pressure head from previous iteration, then this boundary condition only contribute to the modification of the load vector in Eq. (3.3.18). Therefore, the modification of Eq. (3.3.18) is to simply add B_I to L_I.

Variable boundary condition: Dirichlet or Cauchy boundary condition

The implementation of variable-type boundary condition is more involved. During the iteration of boundary conditions on the variable boundary, one of Eqs.(2.3.9) through (2.3.12) is used at a node. If either Eq.(2.3.10) or (2.3.13) is used, we substitute it into Eq.(3.3.6) to yield the value of B_I for the variable node I:

$$B_I = -\int_{B_V} N_I \frac{\rho}{\rho_o} q_p dB, \quad or \quad B_I = -\int_{B_V} N_I \frac{\rho}{\rho_o} q_e dB \tag{3.3.23}$$

which is independent of the pressure head h. Thus, if Eq. (2.3.10) or (2.3.13) is chosen during the iterative process, the implementation of the boundary condition is to simply add B_I to L_I in Eq. (3.3.8) which is the corresponding algebraic equation for boundary node I. On the other hand, if Eq. (2.3.9), (2.3.11), or (2.3.12) is chosen, we override Eq. (3.3.8) with an identity equation as in the implementation of Dirichlet boundary conditions:

$$\begin{gathered} A_{I,1} = 0, .., A_{I,I-1} = 0, A_{I,I} = 1, A_{I,I+1} = 0, .., A_{I,N} = 0 \quad and \\ L_I + B_I = h_p \quad if \quad Eq. \quad (2.3.9)\ is\ used \quad or \\ L_I + B_I = h_p \quad if \quad Eq. \quad (2.3.11)\ is\ used \quad or \\ L_I + B_I = h_m \quad if \quad Eq. \quad (2.3.12)\ is\ used \end{gathered} \tag{3.3.24}$$

River boundary condition:

For the the river boundary condition given by Eq.(2.3.8), we simply substitute Eq.(2.3.8) into Eq.(3.3.6) to yield the following integrals:

$$B_I = \int_{B_r} N_I \frac{\rho}{\rho_o} \frac{K_R}{b_R} h_R dB \quad and \quad B_{I,J} = \int_{B_r} N_I \frac{\rho}{\rho_o} \frac{K_R}{b_R} J_J dB \tag{3.3.25}$$

The integrals B_I and $B_{I,J}$, respectively, are added to L_I and subtracted from $A_{I,J}$, respectively, in Eq. (3.3.18) to complete the modification of this algebraic equation for the node I.

After the incorporation of boundary conditions, we obtain the following matrix equation

$$[C]\{h\} = \{R\} \quad where \quad [C] = [A] + [B] \quad and \quad \{R\} = \{L\} + [\{B\}] \tag{3.3.26}$$

where [C] is the final coefficient matrix; {R} is the final right-hand side vector; and [B] and {B} the coefficient matrix and load vector contributed from boundary conditions. For saturated-unsaturated flow simulations, [C] and {R} are highly nonlinear functions of the pressure head {h}.

Solution of the matrix equation. Equation (3.3.26) is in general a banded sparse matrix equation. It may be solved numerically by either direct method or iteration methods. In direct methods, a sequence of operation is performed only once. This would result in an exact solution except for round-off error. In this method, one is concerned with the efficiency and magnitude of round-off error associated with the sequence of operations. On the other hand, in an iterative method, one attempts to the solution by a process of successive approximations. This involves in making an initial guess, then improving the guess by some iterative process until an error criterion is obtained. Therefore, in this technique, one must be concerned with convergence, and the rate of convergence. The round-off errors tend to be self-corrected.

For practical purposes, the most advantages of direct method are: (1) the efficient computation when the bandwidth of the matrix [C] is small, and (2) the fact that no problem of convergency is encountered when the matrix equation is linear or less severity in convergence than iterative methods even when the matrix equation is nonlinear. The most disadvantages of direct methods are the excessive requirements on CPU storage and CPU time when a large number of nodes is needed for discretization. On the other hand, the most advantages of iterative methods are the efficiencies in terms of CPU storage and CPU time when large problems are encountered. Their most disadvantages are the requirements that the matrix [C] must be well conditioned to guarantee a convergent solution. For three dimensional problems, the bandwidth of the matrix is usually large, thus the direction solution method is not practical. Only the iterative methods are implemented in the three-dimensional flow module of WASH123D. Four iteration methods are used in solving the linearized matrix equation: (1) block iteration, (2) successive point iteration, (3) incomplete Cholesky preconditioned conjugate gradient method, and (4) algebraic multigrid method.

The matrix equation, Eq. (3.326), is nonlinear because both the hydraulic conductivity and the water capacity are functions of the pressure head h. To solve the nonlinear matrix equation, two approaches can be taken: (1) the Picard method and (2) the Newton-Ralphson method. The Newton-Ralphson method has a second order of convergent rate and is very robust. However, the Newton-Ralphson method would destroy the symmetrical property of the coefficient matrix resulting from the finite element approximation. As a result the solution of the linearized matrix equation requires extra care. Many of the iterative methods will not warrant a convergent solution for the non-symmetric linearized matrix equation. Thus, the Picard method is used in this report to solve the nonlinear problems.

In the Picard method, an initial estimate is made of the unknown {h}. Using this estimate, we then compute the coefficient matrix [C] and solve the linearized matrix equation by the method of linear algebra. The new estimate is now obtained by the weighted average of the new solution and the previous estimate:

$$\left\{h^{(k+1)}\right\} = \omega\{h\} + (1-\omega)\left\{h^{k}\right\} \qquad \textbf{(3.3.27)}$$

where $\{h^{(k+1)}\}$ is the new estimate, $\{h^k\}$ is the previous estimate, {h} is the new solution, and ω is the iteration parameter. The procedure is repeated until the new solution {h} is within a tolerance error.

If ω is greater than or equal to 0 but is less than 1, the iteration is under-relaxation. If $\omega = 1$, the method is the exact relaxation. If ω is greater than 1 but less than or equal to 2, the iteration is termed over-relaxation. The under-relaxation should be used to overcome cases when nonconvergency or the slow convergent rate is due to fluctuation rather than due to "blowup" computations. Over-relaxation should be used to speed up convergent rate when it decreases monotonically.

In summary, there are 16 optional numerical schemes here to deal with as wide a range of problems as possible. These are the combinations of: (1) two ways of treating the mass matrix (lumping and no-lumping); (2) two ways of approximating the time derivatives (time-weighting and mid-difference), and (3) four ways of solving the linearized matrix equation.

3.3.2 Numerical Approximations of Thermal Transport Equations

Two options are provided in this report to solve the thermal transport equation. One is the finite element method and the other is the particle tracking method.

3.3.2.1 Finite Element Method. Recall the thermal transport equation is governed by Eq. (2.3.14) that is rewritten in a slightly different form as

$$\left(\rho_W C_W \theta + \rho_b C_m\right)\frac{\partial T}{\partial t} + \frac{\partial\left(\rho_W C_W \theta + \rho_b C_m\right)}{\partial t}T + \nabla\cdot\left(\rho_W C_W \mathbf{V} T\right) - \nabla\cdot\left(\mathbf{D}^{\mathbf{H}} h\cdot\nabla T\right) = H^a + H^C \qquad \textbf{(3.3.28)}$$

Applying the finite element method to Eq. (3.3.28), we obtain the following matrix equation

$$[M]\frac{d\{T\}}{dt} + [V]\{T\} + [D]\{T\} + [K]\{T\} = -\{\Phi^B\} + \{\Phi^a\} + \{\Phi^C\} \qquad \textbf{(3.3.29)}$$

in which

$$M_{ij} = \int_R N_i\left(\rho_W C_W \theta + \rho_b C_m\right)N_j dR, \qquad V_{ij} = \int_R \nabla W_i \rho_W C_W \mathbf{V} N_j dR,$$
$$D_{ij} = \int_R \nabla N_i\cdot\mathbf{D}^{\mathbf{H}}\nabla N_j dR, \qquad K_{ij} = \int_R N_i \frac{\partial\left(\rho_W C_W \theta\right) + \rho_b C_m}{\partial t} N_j dR, \qquad \textbf{(3.3.30)}$$
$$\Phi_i^B = \int_B \mathbf{n}\cdot\left(W_i \rho_W C_W \mathbf{V} T - N_i \mathbf{D}^{\mathbf{H}}\nabla T\right)dB$$

$$\Phi_i^a = \int_R N_i H_a R, \qquad \Phi_i^r = \int_R N_i H_r dR, \qquad \Phi_i^C = \int_R N_i H_C dR \qquad \textbf{(3.3.31)}$$

where W_i is the weighting function of node x_i; N_i and N_j are the base functions of nodes x_i and x_j, respectively; [M] is the mass matrix, [V] is the stiff matrix due to advective transport; [D] is the stiff matrix due to dispersion/diffusion/conduction; {T} is the solution vector of temperature; $\{\Phi^B\}$ is the vector due to boundary conditions, which can contribute to load vector and/or coefficient matrix; $\{\Phi^a\}$ is the load vector due to artificial energy source; $\{\Phi^r\}$ is the load vector due to energy contained in rainfall; and $\{\Phi^c\}$ is the vector due to chemical reaction, which is not considered in this version, but can be added easily.

Approximating the time derivative term in Eq. (3.3.29) with a time-weighted finite difference, we reduce the advective-diffusive equation and its boundary conditions to the following matrix equation

$$[C]\{T\} = \{L\} - \{\Phi^B\} \qquad \textbf{(3.3.32)}$$

in which

$$[C] = \frac{[M]}{\Delta t} + \theta([D]+[K]) + \theta_V[V],$$
$$\{L\} = \left(\frac{[M]}{\Delta t} - (1-\theta)([DS]+[K]) - (1-\theta_V)[V]\right)\{T^{(n)}\} + \{\Phi^a\} + \{\Phi^r\} \qquad \textbf{(3.3.33)}$$

where [C] is the coefficient matrix, {L} is the load vector from initial condition, artificial sink/sources, rainfall, and net radiation; Δt is the time step size; θ is the time weighting factor for the dispersion and linear terms; θ_v is the time weighting factor for the velocity term; and $\{T^{(n)}\}$ is the value of {T} at old time level n. The global boundary conditions must be used to provide $\{\Phi^B\}$ in Eq. (3.3.32).

For a global boundary node I, the corresponding algebraic equation from Eq. (3.3.32) is

$$C_{I1}T_1 + .. + C_{I,I}T_I + .. + C_{I,N}T_N = L_I - \Phi_I^B \qquad \textbf{(3.3.34)}$$

In the above equations there are two unknowns T_I and Φ_I^B; either T_I or Φ_I^B, or the relationship between T_I and Φ_I^B must be specified. The numerical implementation of these boundary conditions is described as follows.

Direchlet boundary condition: prescribed temperature

If T_I is given on the boundary node I (Dirichlet boundary condition), all coefficients ($C_{I,1}$, .., $C_{I,I}$, .., $C_{I,N}$) and the right-hand side term (L_I) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then an identity equation is created as

$$T_I = T_{Idb}, \quad I \in N_D \qquad \textbf{(3.3.35)}$$

where T_{Idb} is the prescribed temperature on the Dirichlet node I and N_D is the number of Dirichlet boundary nodes. This process is repeated for every Dirichlet nodes. Note it is unnecessary to modify other equations that involving these unknowns, which was done in the previous version. By not modifying other equations, the symmetrical property of the matrix is preserved, which makes the iterative solvers more robust. The final set of equations will consist of N_D identity equations and (N - N_D) finite element equations for N unknowns T_i's. After T_i's for all nodes are solved from the matrix equation, Eq. (3.3.34) is then used to back calculate N_D Φ_I^B's.

If a direct solver is used to solve the matrix equation, the above procedure will solve N T_i's accurately except for roundoff errors. However, if an iterative solver is used, a stopping criterion must be strict enough so that the converged solution of N T_i's are accurate enough to the exact solution. With such accurate T_i's, then can be sure that the back-calculated ND Φ_I^B's are accurate.

Cauchy boundary condition: prescribed heat flux

If Φ_{BI} is given (Cauchy flux boundary condition), all coefficients ($C_{I,1}$, .., $C_{I,I}$, .., $C_{I,N}$) and right-hand side term (L_I) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then Eq. (3.3.34) is modified to incorporate the boundary conditions and used to solve for T_I. The modification of Eq. (3.3.34) is straightforward. Because Φ_I^B is a known quantity, it contributes to the load on the right hand side. This type of boundary conditions is very easy to implement. After T_is are obtained, the original Eq. (3.3.34), which is stored in a temporary array, is used to back calculate N_C Φ_I^B's on flux boundaries (where N_C is the number of flux boundary nodes). These back-calculated Φ_I^B's should be theoretically identical to the input Φ_I^B's. However, because of round-off errors (in the case of direct solvers) or because of stopping criteria (in the case of iterative solvers), the back-calculated Φ_I^B's will be slightly different from the input Φ_I^B's. If the differences between the two are significant, it is an indication that the solvers have not yielded accurate solutions.

Neumann boundary condition: prescribed gradient of temperature

At Neumann boundaries, the temperature gradient is prescribed, thus, the flux due to temperature gradient is given. For this case, all coefficients ($C_{I,1}$, .., $C_{I,I}$, .., $C_{I,N}$) and right-hand side term (L_I) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then Eq. (3.3.34) is modified to incorporate the boundary conditions and used to solve for T_I. For the Neumann boundary condition, Φ_I^B contributes to both the matrix coefficient and load vector, thus both the coefficient matrix [C] and the load vector {L} must be modified. Recall

$$\Phi_i^B = \int_B \mathbf{n} \cdot \left(W_i \rho_W C_W \mathbf{V} T - N \mathbf{D}^{\mathbf{H}} \nabla T \right) dB \tag{3.3.36}$$

Substituting Eq. (2.3.19) into Eq. (3.3.36), we have

$$\left\{ \Phi^B \right\} \equiv [CB]\{T\} + \{LB\}$$
$$\textit{in which} \qquad CB_{ij} = -\int_B \mathbf{n} \cdot W_i \rho_W C_W \mathbf{V} N_j dB \quad \textit{and} \quad LB_i = -\int_B N_i \varphi_{nb}(t) dB \tag{3.3.37}$$

where [CB] and {LB} are the coefficient matrix and load vector due to Neumann boundary. Adding the I-th equation in Eq. (3.3.37) to Eq. (3.3.34), we obtained a modified equation, which can be solved for solve T_I. After T_I is solved, the original Eq. (3.3.34) (recall the original Eq. (3.3.34) must be and has been stored in a temporary array) is used to back-calculate Φ_I^B.

Variable boundary condition:

At the variable boundary condition Node I, the implementation of boundary conditions can be made identical to that for a Cauchy boundary condition node if the flow is directed into the region. If the flow is going out of the region, the boundary condition is implemented similar to the implementation of Neuman boundary condition with $LB_I = 0$. The assumption of zero Neumann flux implies that a

Neuman node must be far away from the source/sink.

Atmosphere-subsurface media interface boundary condition:

At the atmosphere-media interface, the heat flux is a nonlinear function of the temperature since the back radiation and the heat flux due to evaporation and sensible heat are both function of temperature. To implement this boundary condition, we first expand Eq. (2.3.20) in Taylor series as follows:

$$-\mathbf{n}\cdot\left(\rho_W C_W \mathbf{V}T - \mathbf{D}^{\mathbf{H}}\cdot\nabla T\right) = F\left(T^{(k)}\right) + \frac{dF}{dT}\Big|_{T=T^{(k)}}\left(T - T^{(k)}\right)$$
$$where\ F = H_n - H_b - H_e - H_s \qquad \textbf{(3.3.38)}$$

where T(k) is the value of T at previous iteration. Substituting Eq. (3.3.38) into Eq. (3.3.36), we have

$$\left\{\Phi^B\right\} \equiv \left[CB\right]\left\{T\right\} + \left\{LB\right\} \quad in\ which$$
$$CB_{ij} = \int_B N_i \frac{dF}{dT}\Big|_{T=T^{(k)}} N_j dB \quad and \quad LB_i = \int_B N_i \left(F\left(T^{(k)}\right) - \frac{dF}{dT}\Big|_{T=T^{(k)}} T^{(k)} \right) dB \qquad \textbf{(3.3.39)}$$

where [CB] and {LB} are the coefficient matrix and load vector due to the atmosphere-media boundary condition. Adding the I-th equation in Eq. (3.3.39) to Eq. (3.3.34), we obtained a modified equation, which can be solved for solve T_I. After T_I is solved, the original Eq. (3.3.34) is used to back-calculate Φ_I^B.

Subsurface-river interface boundary condition:

This type of boundary condition will be addressed in Sub-Sections 3.4.3 and 3.4.4.

Subsurface-overland interface boundary condition:

This type of boundary condition will be addressed in Sub-Section 3.4.2.

3.3.2.2 The Hybrid Lagrangian-Eulerian Finite Element Method. When the hybrid Lagrangian-Eulerian finite element method is used to solve the thermal transport equation, we expand Eq. (3.2.70) to yield following advection-dispersion equation in the Lagrangian form

$$\frac{D_{\mathbf{U}}T}{Dt} + KT = D + \Phi^S \quad where \quad \mathbf{U} = \frac{\rho_W C_r \mathbf{V}}{\left(\rho_W C_W \theta + \rho_b C_m\right)} \qquad \textbf{(3.3.40)}$$

in which

$$K = \frac{1}{\left(\rho_W C_W \theta + \rho_b C_m\right)} \frac{\partial\left(\rho_W C_W \theta + \rho_b C_m\right)}{\partial t} + \frac{1}{\left(\rho_W C_W \theta + \rho_b C_m\right)} \nabla\cdot\left(\rho_W C_W \mathbf{V}\right),$$
$$D = \frac{1}{\left(\rho_W C_W \theta\right) + \rho_b C_m} \nabla\cdot\left(\mathbf{D}^{\mathbf{H}}\cdot\nabla T\right) \quad and \quad \Phi^S = \frac{H^a + H^r}{\left(\rho_W C_W \theta + \rho_b C_m\right)} \qquad \textbf{(3.3.41)}$$

To use the semi-Lagrangian method to solve the thermal transport equation, we integrate Eq. (3.3.40) along its characteristic line from x_i at new time level to x_i^* at old time level or on the boundary, we obtain

$$\left(1+\frac{\Delta\tau}{2}K_i^{(n+1)}\right)T_i^{(n+1)} = \left(1-\frac{\Delta\tau}{2}K_i^{*}\right)T_i^{*} + \frac{\Delta\tau}{2}\left(D_i^{(n+1)}+D_i^{*}\right)+\frac{\Delta\tau}{2}\left(\Phi_i^{S^{(n+1)}}+\Phi_i^{S^{*}}\right),\quad i\in N \tag{3.3.42}$$

where $\Delta\tau$ is the tracking time, it is equal to Δt when the backward tracking is carried out all the way to the root of the characteristic and it is less than Δt when the backward tracking hits the boundary before Δt is consumed; $K_i^{(n+1)}$, $T_i^{(n+1)}$, $D_i^{(n+1)}$, and $\Phi_i^{S(n+1)}$, respectively, are the values of K, T, D, and Φ^S, respectively, at x_i at new time level $t = (n+1)\Delta t$; and K_i^*, T_i^*, D_i^*, and Φ_i^{S*}, respectively, are the values of K, T, D, and Φ^S, respectively, at the location x_i^*.

To compute the dispersion/diffusion terms $D_i^{(n+1)}$ and D_i^*, we rewrite the second equation in Eq. (3.3.41) as

$$\left(\rho_W C_W\theta+\rho_b C_m\right)D = \nabla\cdot\left(\mathbf{D}^{\mathbf{H}}\cdot\nabla T\right) \tag{3.3.43}$$

Applying the Galerkin finite element method to Eq. (3.3.43) at new time level (n+1), we obtain the following matrix equation for $\{D^{(n+1)}\}$ as

$$\left[a^{(n+1)}\right]\left\{D^{(n+1)}\right\}+\left[b^{(n+1)}\right]\left\{T^{(n+1)}\right\}=\left\{B^{(n+1)}\right\} \tag{3.3.44}$$

in which

$$\left\{D^{(n+1)}\right\}=\left\{D_1^{(n+1)}\quad D_2^{(n+1)}\quad .\ .\ D_i^{(n+1)}\quad .\ .\ D_N^{(n+1)}\right\}^{Transpose} \tag{3.3.45}$$

$$\left\{T^{(n+1)}\right\}=\left\{T_1^{(n+1)}\quad T_2^{(n+1)}\quad .\ .\ T_i^{(n+1)}\quad .\ .\ T_N^{(n+1)}\right\}^{Transpose} \tag{3.3.46}$$

$$\left\{B^{(n+1)}\right\}=\left\{B_1^{(n+1)}\quad B_2^{(n+1)}\quad .\ .\ B_i^{(n+1)}\quad .\ .\ B_N^{(n+1)}\right\}^{Transpose} \tag{3.3.47}$$

$$a_{ij}^{(n+1)}=\int_R N_i\left(\rho_W C_W\theta+\rho_b C_m\right)\Big|_{(n+1)}N_j dR,\quad b_{ij}^{(n+1)}=\int_R \nabla N_i\cdot\left(\mathbf{D}^{\mathbf{H}}\right)\Big|_{(n+1)}\cdot\nabla N_j dR,$$
$$B_i^{(n+1)}=\int_B n\cdot N_i\left(D^H\right)\Big|_{(n+1)}\cdot\nabla T^{(n+1)}dB \tag{3.3.48}$$

where the superscript (n+1) denotes the time level; N and N are the base functions of nodes at x_i and x_j, respectively.

Lumping the matrix $[a^{(n+1)}]$, we can solve Eq. (3.3.44) for $D_I^{(n+1)}$ as follows

$$D_I^{(n+1)} = -\frac{1}{a_{II}^{(n+1)}}\sum_j b_{Ij}^{(n+1)} T_j^{(n+1)} \quad if \quad I \text{ is an interior point}$$
$$D_I^{(n+1)} = \frac{1}{a_{II}^{(n+1)}} B_I^{(n+1)} - \frac{1}{a_{II}^{(n+1)}}\sum_j b_{Ij}^{(n+1)} T_j^{(n+1)} \quad if \quad I \text{ is a boundary point} \tag{3.3.49}$$

where $a_{II}^{(n+1)}$ is the lumped $a_{ii}^{(n+1)}$. Following the identical procedure that leads Eq. (3.3.43) to Eq. (3.3.49), we have

$$D_I^{(n)} = -\frac{1}{a_{II}^{(n)}}\sum_j b_{Ij}^{(n)} T_j^{(n)} \quad if \quad I \text{ is an interior point}$$
$$D_I^{(n)} = \frac{1}{a_{II}^{(n)}} B_I^{(n)} - \frac{1}{a_{II}^{(n)}}\sum_j b_{Ij}^{(n)} T_j^{(n)} \quad if \quad I \text{ is a boundary point} \tag{3.3.50}$$

where $\{B^{(n)}\}$, $\{a^{(n)}\}$ and $\{b^{(n)}\}$, respectively, are defined similar to $\{B^{(n+1)}\}$, $\{a^{(n+1)}\}$ and $\{b^{(n+1)}\}$, respectively.

With $\{D^{(n)}\}$ calculated with Eq. (3.3.50), $\{D^*\}$ can be interpolated. Substituting Eq. (3.3.49) into Eq. (3.3.42) and implementing boundary conditions given in Section 2.3.2, we obtain a system of N simultaneous algebraic equations N unknowns ($T_i^{(n+1)}$ for i = 1, 2, .., N.) If the dispersion/diffusion term is not included, then Eq. (3.3.42) is reduced to a set of N decoupled equations as

$$a_{ii} T_i^{(n+1)} = b_i, \quad i \in N \tag{3.3.51}$$

where

$$a_{ii} = \left(1 + \frac{\Delta\tau}{2} K_i^{(n+1)}\right)$$
$$b_i = \left(1 - \frac{\Delta\tau}{2} K_i^{\,*}\right) T_i^{\,*} + \frac{\Delta\tau}{2}\left(\Phi_i^{S(n+1)} + \Phi_i^{S*}\right), \quad i \in N \tag{3.3.52}$$

Equations (3.3.51) is applied to all interior nodes without having to make any modification. On a boundary point, there are two possibilities: Eq. (3.3.51) is replaced with a boundary equation when the flow is directed into the region or Eq. (3.3.51) is still valid when the flow is direct out of the region. In other words, when the thermal energy is transported out of the region at a boundary node (i.e., when $\mathbf{N}\cdot\mathbf{V} \geq 0$), a boundary condition is not needed and Equation (3.3.51) is used to compute the $T_i^{(n+1)}$. When the thermal energy is transported into the region at a node (i.e., when $\mathbf{N}\cdot\mathbf{V} < 0$), a boundary condition must be specified.

Alternatively, to facilitate the implementation of boundary condition at incoming flow node, the algebraic equation for the boundary node is obtained by applying the finite element method to the boundary node. For this alternative approach, the implementation of boundary conditions at global boundary nodes is identical to that in the finite element approximation of solving the thermal transport equation.

3.3.3 Numerical Approximations of Salinity Transport

Two options are provided in this report to solve the salinity transport equation. One is the finite element method and the other is the particle tracking method.

3.3.3.1 Finite Element Method. Recall the salinity transport equation is governed by Eq. (2.3.23) which is rewritten in a slightly different form as

$$\theta \frac{\partial S}{\partial t} + \frac{\partial \theta}{\partial t} S + \nabla \cdot (\mathbf{V} S) - \nabla \cdot (\theta \mathbf{D} \cdot \nabla h) = S^{as} \tag{3.3.53}$$

Applying the finite element method to Eq. (3.3.53), we obtain the following matrix equation

$$[M]\frac{d\{S\}}{dt} + [V]\{S\} + [D]\{S\} + [K]\{S\} = -\{\Psi^B\} + \{\Psi^a\} \tag{3.3.54}$$

in which

$$\begin{aligned} &M_{ij} = \int_R N_i \theta N_j dx, \qquad V_{ij} = \int_R W_i \cdot \mathbf{V} N_j dR, \qquad D_{ij} = \int_R \nabla N_i \cdot \theta \mathbf{D} \cdot \nabla N_j dR, \\ &K_{ij} = \int_R N_i \frac{\partial \theta}{\partial t} N_j dR, \qquad \Psi_i^B = \int_B \mathbf{n} \cdot (W_i \mathbf{V} S - N_i \theta \mathbf{D} \cdot \nabla S) dB, \qquad \Psi_i^a = \int_R N_i S^{as} dR \end{aligned} \tag{3.3.55}$$

$$\Psi_i^a = \int_R N_i M_s^{as} dR, \qquad \Psi_i^r = \int_R N_i M_s^{rs} dR, \qquad \Psi_i^e = \int_R N_i M_s^{es} dR, \qquad \Psi_i^i = \int_R N_i M_s^{is} dR \tag{3.3.56}$$

where W_i is the weighting function of node x_i; N_i and N_j are the base functions of nodes x_i and x_j, respectively; [M] is the mass matrix, [V] is the stiff matrix due to advective transport; [D] is the stiff matrix due to dispersion/diffusion/conduction; [K] is the stiff matrix due to the linear term; {S} is the solution vector of salinity; $\{\Psi^B\}$ is the vector due to boundary conditions, which can contribute to load vector and/or coefficient matrix; and $\{\Psi^a\}$ is the load vector due to artificial salt source.

Approximating the time derivative term in Eq. (3.3.54) with a time-weighted finite difference, we reduce the advective-diffusive equation and its boundary conditions to the following matrix equation.

$$[C]\{S\} = \{L\} - \{\Psi^B\} \tag{3.3.57}$$

in which

$$\begin{aligned} &[C] = \frac{[M]}{\Delta t} + \theta([D] + [K]) + \theta_V [V], \\ &\{L\} = \left(\frac{[M]}{\Delta t} - (1-\theta)([D] + [K]) - (1-\theta_V)[V]\right)\{S^{(n)}\} + \{\Psi^a\} \end{aligned} \tag{3.3.58}$$

where [C] is the coefficient matrix, {L} is the load vector from initial condition, artificial sink/sources and rainfall; Δt is the time step size; θ is the time weighting factor for the dispersion and linear terms; θ_v is the time weighting factor for the velocity term; and $\{S^{(n)}\}$ is the value of {S} at old time level n. The global boundary conditions must be used to provide $\{\Psi^B\}$ in Eq. (3.3.57).

For a global boundary node I, the corresponding algebraic equation from Eq. (3.3.57) is

$$C_{I,1}S_1 + .. + C_{I,I}S_I + .. + C_{I,N}S_N = L_I - \Psi_I^B \quad \textbf{(3.3.59)}$$

In the above equations there are two unknowns T_I and Ψ_I^B; either T_I or Ψ_I^B, or the relationship between T_I and Ψ_I^B must be specified. The numerical implementation of these boundary conditions are described as follows.

Dirichlet boundary condition: prescribed salinity

If S_I is given on the boundary node I (Dirichlet boundary condition), all coefficients ($C_{I,1}$, .., $C_{I,I}$, .., $C_{I,N}$) and the right-hand side term (L_I) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then an identity equation is created as

$$S_I = S_{Id}, \quad I \in N_D \quad \textbf{(3.3.60)}$$

where S_{Id} is the prescribed salinity on the Dirichlet node I and N_D is the number of Dirichlet boundary nodes. This process is repeated for every Dirichlet nodes. Note it is unnecessary to modify other equations that involving this unknowns, which was done in the previous version. By not modifying other equations, the symmetrical property of the matrix is preserved, which makes the iterative solvers more robust. The final set of equations will consist of N_D identity equations and (N - N_D) finite element equations for N unknowns S_i's. After S_i's for all nodes are solved from the matrix equation, Eq. (3.3.59) is then used to back calculate N_D Ψ_I^B's.

If a direct solver is used to solve the matrix equation, the above procedure will solve N S_i's accurately except for roundoff errors. However, if an iterative solver is used, a stopping criterion must be strict enough so that the converged solution of N S_i's are accurate enough to the exact solution. With such accurate S_i's, then can we be sure that the back-calculated ND Ψ_{BI}'s are accurate.

Cauchy boundary condition: prescribed salt flux

If Ψ_I^B is given (Cauchy flux boundary condition), all coefficients ($C_{I,1}$, .., $C_{I,I}$, .., $C_{I,N}$) and the right-hand side term (L_I) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then Eq. (3.3.59) is modified to incorporate the boundary conditions and used to solve for S_I. The modification of Eq. (3.3.59) is straightforward. Because Ψ_I^B is a known quantity, it contributes to the load on the right hand side. This type of boundary conditions is very easy to implement. After S_i's are obtained, the original Eq. (3.3.59), which is stored in a temporary array, is used to back calculate N_C Ψ_I^B's on flux boundaries (where N_C is the number of flux boundary nodes). These back-calculated Ψ_I^B's should be theoretically identical to the input Ψ_I^B's. However, because of round-off errors (in the case of direct solvers) or because of stopping criteria (in the case of iterative solvers), the back-calculated Ψ_I^B's will be slightly different from the input Ψ_I^B's. If the differences between the two are significant, it is an indication that the solvers have not yielded accurate solutions.

Neumann boundary condition: prescribed gradient of salinity

At Neumann boundaries, the temperature gradient is prescribed, thus, the flux due to temperature gradient is given. For this case, all coefficients ($C_{I,}$, .., $C_{I,I}$, .., $C_{I,N}$) and the right-hand side term (L_I) obtained before the implementation of boundary conditions for this equation are stored in a temporary array, then Eq. (3.3.59) is modified to incorporate the boundary conditions and used to solve for S_I. For the Neumann boundary condition, Ψ_I^B contributes to both the matrix coefficient and load vector, thus both the coefficient matrix [C] and the load vector {L} must be modified. Recall

$$\Psi_i^B = \int_B \mathbf{n}\cdot\left(W_i\mathbf{V}S - N_i\theta\mathbf{D}\nabla S\right)dB \tag{3.3.61}$$

Substituting Eq. (2.3.28) into Eq. (3.3.61), we have

$$\begin{aligned}&\left\{\Psi^B\right\} \equiv [CB]\{S\} + \{LB\}\\ &\textit{in which}\qquad CB_{i,j} = \int_B \mathbf{n}\cdot W_i\mathbf{V}N_j dB \quad \textit{and} \quad LB_i = \int_B N_i Q_{Snb}(t)dB\end{aligned} \tag{3.3.62}$$

where [CB] and {LB} are the coefficient matrix and load vector due to Neumann boundary. Adding the I-th equation in Eq. (3.3.62) to Eq. (3.3.59), we obtained a modified equation, which can be solved for solve S_I. After S_I is solved, the original Eq. (3.3.59) is used to back-calculate Ψ_I^B.

Variable boundary condition:

At the variable boundary condition Node I, the implementation of boundary conditions can be made identical to that for a Cauchy boundary condition node if the flow is directed into the river/stream/canal reach. If the flow is going out of the reach, the boundary condition is implemented similar to the implementation of Neuman boundary condition with $\Psi_I^{nb} = 0$. The assumption of zero Neumann flux implies that a Neuman node must be far away from the source/sink.

Subsurface-river interface boundary condition:

This type of boundary condition will be addressed in Sub-Sections 3.4.3 and 3.4.4.

Subsurface-overland interface boundary condition:

This type of boundary condition will be addressed in Sub-Section 3.4.2.

3.3.3.2 The Hybrid Lagrangian-Eulerian Finite Element Method. When the hybrid Lagrangian-Eulerian finite element method is used to solve the salt transport equation, we expand Eq. (3.3.53) to yield following advection-dispersion equation in the Lagrangian form

$$\frac{D_U S}{Dt} + KS = D + \Psi^S \quad \textit{where} \quad \mathbf{U} = \frac{\mathbf{V}}{\theta} \tag{3.3.63}$$

in which

$$K = \frac{1}{\theta}\frac{\partial\theta}{\partial t} + \frac{1}{\theta}\nabla\cdot(\mathbf{V}), \quad D = \frac{1}{\theta}\nabla\cdot\left(\theta\mathbf{D}\cdot\nabla S\right) \quad \textit{and} \quad \Psi^S = \frac{S^{as}}{\theta} \tag{3.3.64}$$

To use the semi-Lagrangian method to solve the thermal transport equation, we integrate Eq. (3.3.63) along its characteristic line from x_i at new time level to x_i^* at old time level or on the boundary, we obtain

$$\left(1+\frac{\Delta\tau}{2}K_i^{(n+1)}\right)S_i^{(n+1)} = \left(1-\frac{\Delta\tau}{2}K_i^*\right)S_i^* + \frac{\Delta\tau}{2}\left(D_i^{(n+1)}+D_i^*\right)+\frac{\Delta\tau}{2}\left(\Psi_i^{S^{(n+1)}}+\Psi_i^{S*}\right),\quad i\in N \qquad \textbf{(3.3.65)}$$

where $\Delta\tau$ is the tracking time, it is equal to Δt when the backward tracking is carried out all the way to the root of the characteristic and it is less than Δt when the backward tracking hits the boundary before Δt is consumed; $K_i^{(n+1)}$, $T_i^{(n+1)}$, $D_i^{(n+1)}$, and $\Psi_i^{S(n+1)}$, respectively, are the values of K, T, D, and Ψ^S, respectively, at x_i at new time level $t = (n+1)\Delta t$; and K_i^*, T_i^*, D_i^*, and Ψ_i^{S*}, respectively, are the values of K, T, D, and Ψ^S, respectively, at the location x_i^*.

To compute the dispersion/diffusion terms $D_i^{(n+1)}$ and D_i^*, we rewrite the second equation in Eq. (3.3.64) as

$$\theta D = \nabla\cdot\left(\theta\mathbf{D}\cdot\nabla S\right) \qquad \textbf{(3.3.66)}$$

Applying the Galerkin finite element method to Eq. (3.3.66) at new time level (n+1), we obtain the following matrix equation for $\{D^{(n+1)}\}$ as

$$\left[a^{(n+1)}\right]\left\{D^{(n+1)}\right\}+\left[b^{(n+1)}\right]\left\{S^{(n+1)}\right\}=\left\{B^{(n+1)}\right\} \qquad \textbf{(3.3.67)}$$

in which

$$\left\{D^{(n+1)}\right\}=\left\{D_1^{(n+1)}\quad D_2^{(n+1)}\quad .\ .\ D_i^{(n+1)}\ .\ .\ D_N^{(n+1)}\right\}^{Transpose} \qquad \textbf{(3.3.68)}$$

$$\left\{S^{(n+1)}\right\}=\left\{S_1^{(n+1)}\quad S_2^{(n+1)}\quad .\ .\ S_i^{(n+1)}\ .\ .\ S_N^{(n+1)}\right\}^{Transpose} \qquad \textbf{(3.3.69)}$$

$$\left\{B^{(n+1)}\right\}=\left\{B_1^{(n+1)}\quad B_2^{(n+1)}\quad .\ .\ B_i^{(n+1)}\ .\ .\ B_N^{(n+1)}\right\}^{Transpose} \qquad \textbf{(3.3.70)}$$

$$a_{ij}^{(n+1)}=\int_R N_i\left(\theta\right)\Big|_{(n+1)}N_j dR,\quad b_{ij}^{(n+1)}=\int_R \nabla N_i\cdot\left(\theta\mathbf{D}\right)\Big|_{(n+1)}\cdot\nabla N_j dR,$$
$$B_i^{(n+1)}=\int_B n\cdot N_i\left(\theta\mathbf{D}\right)\Big|_{(n+1)}\cdot\nabla S^{(n+1)}dB \qquad \textbf{(3.3.71)}$$

where the superscript (n+1) denotes the time level; N and N are the base functions of nodes at x_i and x_j, respectively.

Lumping the matrix $[a^{(n+1)}]$, we can solve Eq. (3.2.110) for $D_I^{(n+1)}$ as follows

$$D_I^{(n+1)} = -\frac{1}{a_{II}^{(n+1)}}\sum_j b_{Ij}^{(n+1)} S_j^{(n+1)} \quad if \quad I \text{ is an interior point}$$
$$D_I^{(n+1)} = \frac{1}{a_{II}^{(n+1)}} B_I^{(n+1)} - \frac{1}{a_{II}^{(n+1)}}\sum_j b_{Ij}^{(n+1)} S_j^{(n+1)} \quad if \quad I \text{ is a boundary point} \qquad \textbf{(3.3.72)}$$

where $a_{II}^{(n+1)}$ is the lumped $a_{ii}^{(n+1)}$. Following the identical procedure that leads Eq. (3.3.66) to Eq. (3.3.72), we have

$$D_I^{(n)} = -\frac{1}{a_{II}^{(n)}}\sum_j b_{Ij}^{(n)} S_j^{(n)} \quad if \quad I \text{ is an interior point}$$
$$D_I^{(n)} = \frac{1}{a_{II}^{(n)}} B_I^{(n)} - \frac{1}{a_{II}^{(n)}}\sum_j b_{Ij}^{(n)} S_j^{(n)} \quad if \quad I \text{ is a boundary point} \qquad \textbf{(3.3.73)}$$

where $\{B^{(n)}\}$, $\{a^{(n)}\}$ and $\{b^{(n)}\}$, respectively, are defined similar to $\{B^{(n+1)}\}$, $\{a^{(n+1)}\}$ and $\{b^{(n+1)}\}$, respectively.

With $\{D^{(n)}\}$ calculated with Eq. (3.3.73), $\{D^*\}$ can be interpolated. Substituting Eq. (3.3.72) into Eq. (3.3.65) and implementing boundary conditions given in Section 2.3.3, we obtain a system of N simultaneous algebraic equations N unknowns ($S_i^{(n+1)}$ for i = 1, 2, .., N.) If the dispersion/diffusion term is not included, then Eq. (3.3.65) is reduced to a set of N decoupled equations as

$$a_{ii} S_i^{(n+1)} = b_i, \quad i \in N \qquad \textbf{(3.3.74)}$$

where

$$a_{ii} = \left(a + \frac{\Delta\tau}{2} K_i^{(n+1)}\right), \quad b_i = \left(1 - \frac{\Delta\tau}{2} K_i^*\right) S_i^* + \frac{\Delta\tau}{2}\left(\Psi_i^{S^{(n+1)}} + \Psi_i^{S^*}\right), \quad i \in N \qquad \textbf{(3.3.75)}$$

Equations (3.3.75) is applied to all interior nodes without having to make any modification. On a boundary point, there two possibilities: Eq. (3.3.75) is replaced with a boundary equation when the flow is directed into the region or Eq. (3.3.75) is still valid when the flow is direct out of the region. In other words, when the salt is transported out of the region at a boundary node (i.e., when **N•V** ≥ 0), a boundary condition is not needed and Equation (3.3.75) is used to compute the $S_i^{(n+1)}$. When the salt is transported into the region at a node (i.e., when **N•V** < 0), a boundary condition must be specified.

Alternatively, to facilitate the implementation of boundary condition at incoming flow node, the algebraic equation for the boundary node is obtained by applying the finite element method to the boundary node. For this alternative approach, the implementation of boundary conditions at global boundary nodes is identical to that in the finite element approximation of solving the salt transport equation.

3.4 Numerical Implementation of Flow Coupling among Various Media

This section addresses numerical implement of coupling flow simulations among various media including (1) between 1D river and 2D overland flows, (2) between 2D overland and 3D subsurface flows, (3) between 3D subsurface and 1D overland flows, and (4) among 1D river, 2D overland, and 3D subsurface flows. Without loss of generality, numerical implementations of coupling for water flow equations are heuristically given for finite element approximations of diffusive wave models. For Largrangian-Eulerian approximations of diffusive wave models, semi-Largrangian approximations of kinematic wave models, or particle tracking approximations of fully dynamic wave models in surface waters, the implementations of numerical coupling among various media remain valid.

3.4.1 Coupling between 1-D River Networks and 2-D Overland Flows

The interaction between one-dimensional river and two-dimensional overland flows involves two cases: one is between overland and river nodes (left frame in Fig. 3.4-1) and the other is between overland and junction nodes (right frame in Fig. 3.4-1). For every river node (Node *I* in the left frame of Fig. 3.4-1), there will be associated with two overland nodes (Nodes *J* and *K* in the left frame of Fig. 3.4-1). For every junction node (Node *L* in the right frame of Fig. 3.4-1), there will be associated with a number of overland nodes such as Nodes *J*, *K*, *O*, etc (right frame of Fig. 3.4-1). It should be noted that nodes, such as Nodes *J* and *K* in the right frame of Figure 3.4-1, contribute flow to both the river as source/sink of Node *I* and the Junction as source/sink of Node *L*.

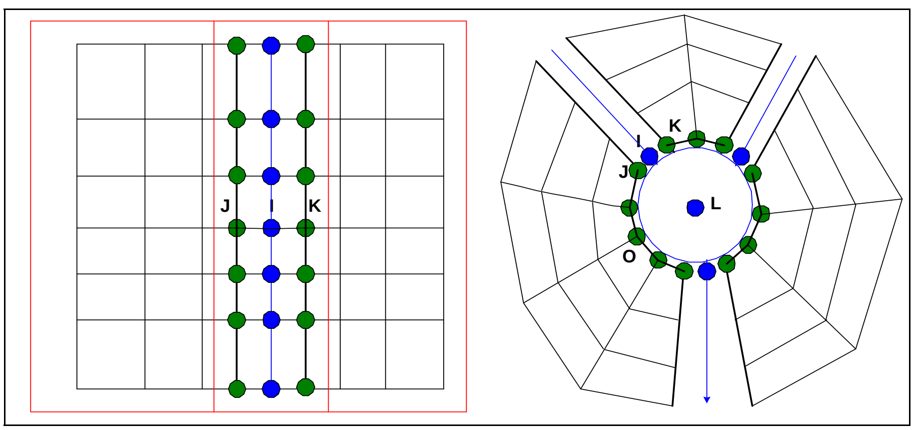

Fig. 3.4-1. Depiction of Interacting River Nodes and Overland Nodes (left) and Junction Nodes and Overland Nodes (Right)

3.4.1.1 Couple Flow Rates between the River Network and the Overland Regime.

Numerical approximations of the diffusive water flow equation for one-dimensional river with finite element methods yield the following matrix

$$
\begin{bmatrix}
-- & -- & -- & -- & -- & -- & -- \\
-- & -- & -- & -- & -- & -- & -- \\
-- & -- & -- & -- & -- & -- & -- \\
A_{I1}^{c} & A_{I2}^{c} & -- & A_{I1}^{c} & -- & -- & A_{IN}^{c} \\
-- & -- & -- & -- & -- & -- & -- \\
-- & -- & -- & -- & -- & -- & -- \\
-- & -- & -- & -- & -- & -- & --
\end{bmatrix}
\begin{Bmatrix} H_1^c \\ H_2^c \\ -- \\ H_I^c \\ -- \\ -- \\ H_N^c \end{Bmatrix}
=
\begin{Bmatrix} R_1^c \\ R_2^c \\ -- \\ R_I^c \\ -- \\ -- \\ R_N^c \end{Bmatrix}
+
\begin{Bmatrix} Q_1^{o1} \\ Q_2^{o1} \\ -- \\ Q_I^{o1} \\ -- \\ -- \\ Q_N^{o1} \end{Bmatrix}
+
\begin{Bmatrix} Q_1^{o2} \\ Q_2^{o2} \\ -- \\ Q_I^{o2} \\ -- \\ -- \\ Q_N^{o2} \end{Bmatrix}
\qquad \textbf{(3.4.1)}
$$

where the superscript c denotes the canal (channel, river, or stream); A_{IJ} is the I-th row, J-th column of the coefficient matrix $[A]$; H_I denotes the water surface at Node I; R_I is I-th entry of the load vector $\{R\}$; N is the number of nodes in the canal; Q_I is the rates of water source/sink from/to the overland flow to/from canal node I; and the superscripts, $o1$ and o2, respectively, denote canal bank 1 and 2, respectively. Every canal node I involves 3 unknowns, H_I^c, Q_I^{o1}, and Q_I^{o2}. However, Eq. (3.4.1) gives just one algebraic equation for every canal node I. Clearly, two additional algebraic equations are need for every canal node I.

Applications of finite element methods to two-dimensional diffusive wave flow equations yield the following matrix

$$
\begin{bmatrix}
A_{11}^{o} & A_{12}^{o} & -- & -- & -- & -- & A_{1M}^{o} \\
A_{21}^{o} & -- & -- & -- & -- & -- & A_{2M}^{o} \\
-- & -- & -- & -- & -- & -- & -- \\
A_{J1}^{o} & A_{J2}^{o} & -- & A_{JJ}^{o} & -- & -- & A_{1M}^{o} \\
-- & -- & -- & -- & -- & -- & -- \\
A_{K1}^{o} & A_{K2}^{o} & -- & -- & A_{KK}^{o} & -- & A_{KM}^{o} \\
-- & -- & -- & -- & -- & -- & -- \\
A_{M1}^{o} & A_{M2}^{o} & -- & -- & -- & -- & A_{MM}^{o}
\end{bmatrix}
\begin{Bmatrix} H_1^o \\ H_2^o \\ -- \\ H_J^o \\ -- \\ H_K^o \\ -- \\ H_M^o \end{Bmatrix}
=
\begin{Bmatrix} R_1^o \\ R_2^o \\ -- \\ R_J^o \\ -- \\ R_K^o \\ -- \\ R_M^o \end{Bmatrix}
-
\begin{Bmatrix} -- \\ -- \\ -- \\ Q_J^o \\ -- \\ Q_K^o \\ -- \\ -- \end{Bmatrix}
\qquad \textbf{(3.4.2)}
$$

where the superscript o denotes the overland; A_{IJ} is the I-th row, J-th column of the coefficient matrix $[A]$; H_I denotes the water surface at Node I; R_I is I-th entry of the load vector $\{R\}$; M is the number of nodes in the overland ; and Q_J and Q_K are the rates of water sink/source from/to the overland to/from the canal via nodes J and K, respectively. Equation (3.4.2) indicates that there is one unknown corresponding to one algebraic equation for every interior node. However, for every algebraic equation corresponding an overland-canal interface node, there are two unknowns, the water surface and the flow rate. Therefore, for every overland-river interface node, one additional equation is needed. Since for every canal node, there are associated two overland-interface nodes, four additional equations are needed for every canal node I for the four additional unknowns Q_I^o, Q_K^o, Q_I^{o1}, and Q_I^{o2}.

The additional equations are obtained by two interface boundary conditions. The first one is the continuity of flux. The second one is the imposition of continuity of water surfaces between canal and overland nodes or the formulation of flow rates. Two of the additional equations are obtained from the interface condition between the canal node *I* and the overland node *J* as

$$Q_J^o = Q_I^{o1}; \qquad H_J^o = H_I^c \quad or \quad Q_I^{o1} = f_1\left(h_J^o, h_I^c\right) \qquad \textbf{(3.4.3)}$$

where f_1 is a prescribed function of water depths h_J^o and h_I^c at the overland node *J* and the canal node *I*. The other two additional equations are obtained from the interface condition between the canal node *I* and the overland node *K*

$$Q_K^o = Q_I^{o2}; \qquad H_K^o = H_I^c \quad or \quad Q_I^{o2} = f_2\left(h_K^o, h_I^c\right) \qquad \textbf{(3.4.4)}$$

where f_2 is a prescribed function of water depths h_K^o and h_I^c at the overland node *K* and the canal node *I*.

When the direct contribution of flow from the overland regime to a junction node *L* (Fig. 3.4-1) is significant, Equations (3.1.77) or (3.1.78) must be modified

$$\frac{d\mathcal{V}_L}{dh_L}\frac{dh_L}{dt} = \sum_{i=1}^{i=3} Q_{iL}^i + \sum_{O \in N_O} Q_O^o \qquad \textbf{(3.4.5)}$$

or

$$\sum_{i=1}^{i=3} Q_{iL}^i + \sum_{O \in N_O} Q_O^o = \sum_{i=1}^{i=3} V_{iL}^i A_{iL}^i + \sum_{O \in N_O} Q_O^o = 0 \qquad \textbf{(3.4.6)}$$

where h_L and $\mathcal{V}_L$ are the water depth and volume at the junction node *L*, Q_{iL}^i is the flux contributed from the node *iL* of the reach *i*, Q_O^o is the flux contributed from the overland node *O* to the junction and N_O is the number of overland nodes interfacing with the junction *L*. Additional N_O unknowns have been introduced in Equation (3.4.5) or (3.4.6). For each overland-junction interface node, say *O* (the right frame in Fig. 3.4-1), the finite element equation written out of Eq. (3.4.2) is

$$A_{O1}^o H_1^o + A_{O2}^o H_2^o + .. + A_{OO}^o H_O^o + .. + A_{OM}^o H_M^o = R_O^o - Q_O^o \qquad \textbf{(3.4.7)}$$

It is seen that Equation (3.4.7) involves two unknowns, H_O^o and Q_O^o. One equation must be supplemented to the finite element equation to close the system. This equation is obtained by either imposing the continuity of water surfaces between nodes *O* and *L* or formulating flux as

$$H_O^o = H_L \quad or \quad Q_O^o = f_o\left(h_O^o, h_L\right) \qquad \textbf{(3.4.8)}$$

where f_o is a prescribed function of water depths at nodes *O* and *L*.

Finally, for each reach-junction interface node, say node *I* (the right frame in Fig. 3.4-1) which we shall say Node *1L* of the first reach connecting to Junction *L*, the formulation of Q_{1L}^1 (or Q_I^1) is similar to that of Equation (3.4.9) as

$$H_I^1 = H_L \quad or \quad Q_I^1 = f_1\left(h_I^1, h_L\right) \qquad \textbf{(3.4.9)}$$

where the superscript *1* denotes reach number and the subscript *I* denote node number.

3.4.1.1 Couple thermal or Salt Rate between the River Network and the Overland Regime.

Numerical approximations of thermal or salt transport equation for one-dimensional river with finite element methods yield the following matrix

$$\begin{bmatrix} -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ C_{I1}^c & C_{I2}^c & -- & C_{I1}^c & -- & -- & C_{IN}^c \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \end{bmatrix} \begin{Bmatrix} E_1^c \\ E_2^c \\ -- \\ E_I^c \\ -- \\ -- \\ E_N^c \end{Bmatrix} = \begin{Bmatrix} R_1^c \\ R_2^c \\ -- \\ R_I^c \\ -- \\ -- \\ R_N^c \end{Bmatrix} + \begin{Bmatrix} M_1^{o1} \\ M_2^{o1} \\ -- \\ M_I^{o1} \\ -- \\ -- \\ M_N^{01} \end{Bmatrix} + \begin{Bmatrix} M_1^{o2} \\ M_2^{o2} \\ -- \\ M_I^{o2} \\ -- \\ -- \\ M_N^{o2} \end{Bmatrix} \qquad \textbf{(3.4.10)}$$

where the superscript *c* denotes the canal (channel, river, or stream); C_{IJ} is the I-th row, J-th column of the coefficient matrix [*C*]; E_I denotes the temperature or salinity at Node *I*; R_I is *I*-th entry of the load vector {*R*}; *N* is the number of nodes in the canal; M_I is the rate of energy or salt source/sink from/to the overland flow to/from canal node *I*; and the superscripts, *o1* and o2, respectively, denote canal bank 1 and 2, respectively. Every canal node *I* involves 3 unknowns, E_I^c, M_I^{o1}, and M_I^{o2}. However, Eq. (3.4.10) gives just one algebraic equation for every canal node *I*. Clearly, two additional algebraic equations are need for every canal node *I*.

Applications of finite element methods to two-dimensional thermal or salt transport equation yield the following matrix

$$\begin{bmatrix} C_{11}^o & C_{12}^o & -- & -- & -- & -- & C_{1M}^o \\ C_{21}^o & -- & -- & -- & -- & -- & C_{2M}^o \\ -- & -- & -- & -- & -- & -- & -- \\ C_{J1}^o & C_{J2}^o & -- & C_{jj}^o & -- & -- & C_{IM}^o \\ -- & -- & -- & -- & -- & -- & -- \\ C_{K1}^o & C_{K2}^o & -- & -- & C_{KK}^o & -- & C_{KM}^o \\ -- & -- & -- & -- & -- & -- & -- \\ C_{M1}^o & C_{M2}^o & -- & -- & -- & -- & C_{MM}^o \end{bmatrix} \begin{Bmatrix} E_1^o \\ E_2^o \\ -- \\ E_J^o \\ -- \\ E_K^o \\ -- \\ E_M^o \end{Bmatrix} = \begin{Bmatrix} R_1^o \\ R_2^o \\ -- \\ R_J^o \\ -- \\ R_K^o \\ -- \\ R_M^o \end{Bmatrix} - \begin{Bmatrix} -- \\ -- \\ -- \\ M_J^o \\ -- \\ M_K^o \\ -- \\ -- \end{Bmatrix} \qquad \textbf{(3.4.11)}$$

where the superscript *o* denotes the overland; C_{IJ} is the *I*-th row, *J*-th column of the coefficient matrix [*C*]; E_I denotes the temperature or salinity at Node *I*; R_I is *I*-th entry of the load vector {*R*}; *M* is the number of nodes in the overland; and M_J and M_K are the rates of thermal or salt sink/source from/to the overland to/from the canal via nodes *J* and *K*, respectively. Equation (3.4.11) indicates that there is one unknown corresponding to one algebraic equation for every interior node. However, for every algebraic equation corresponding to an overland-canal interface node, there are two unknowns, the temperature or salinity and the thermal or salt flux. Therefore, for every overland-river interface node, one additional equation is needed. Since for every canal node, there are associated two overland-interface nodes, four additional equations are needed for every canal node *I* for the four additional unknowns M_I^o, M_K^o, M_I^{o1}, and M_I^{o2}.

The additional equations are obtained by two interface boundary conditions. The first one is the continuity of flux. The second one is the assumption that the thermal or salinity rates through the interface node are due mainly to water flow (i.e., advection). Two of the additional equations are obtained from the interface condition between the canal node *I* and the overland node *J* as

$$\begin{aligned} M_I^{o1} &= \rho_W C_W Q_I^{o1} \frac{1}{2}\left(\left(1+sign\left(Q_I^{o1}\right)\right)E_J^o + \left(1-sign\left(Q_I^{o1}\right)\right)E_I^c\right) \quad and \\ M_J^o &= \rho_W C_W Q_J^o \frac{1}{2}\left(\left(1+sign\left(Q_J^o\right)\right)E_J^o + \left(1-sign\left(Q_J^o\right)\right)E_I^c\right) \end{aligned} \qquad \textbf{(3.4.12)}$$

for thermal transport or

$$\begin{aligned} M_I^{o1} &= Q_I^{o1} \frac{1}{2}\left(\left(1+sign\left(Q_I^{o1}\right)\right)E_J^o + \left(1-sign\left(Q_I^{o1}\right)\right)E_I^c\right) \quad and \\ M_J^o &= Q_J^o \frac{1}{2}\left(\left(1+sign\left(Q_J^o\right)\right)E_J^o + \left(1-sign\left(Q_J^o\right)\right)E_I^c\right) \end{aligned} \qquad \textbf{(3.4.13)}$$

for salt transport. It should be noted that in Equations (3.4.12) and (3.4.13) $Q_I^{o1} = Q_J^o$, thus the continuity $M_I^{o1} = M_J^o$ is preserved.

The other two additional equations are obtained from the interface condition between the canal node I and the overland node K as follows.

$$\begin{aligned} M_I^{o2} &= \rho_W C_W Q_I^{o2} \frac{1}{2}\left(\left(1+sign\left(Q_I^{o2}\right)\right)E_K^o + \left(1-sign\left(Q_I^{o2}\right)\right)E_I^c\right) \quad and \\ M_K^o &= \rho_W C_W Q_K^o \frac{1}{2}\left(\left(1+sign\left(Q_K^o\right)\right)E_K^o + \left(1-sign\left(Q_K^o\right)\right)E_I^c\right) \end{aligned} \qquad \textbf{(3.4.14)}$$

for thermal transport or

$$\begin{aligned} M_I^{o2} &= Q_I^{o2} \frac{1}{2}\left(\left(1+sign\left(Q_I^{o2}\right)\right)E_K^o + \left(1-sign\left(Q_I^{o2}\right)\right)E_I^c\right) \quad and \\ M_K^o &= Q_K^o \frac{1}{2}\left(\left(1+sign\left(Q_K^o\right)\right)E_K^o + \left(1-sign\left(Q_K^o\right)\right)E_I^c\right) \end{aligned} \qquad \textbf{(3.4.15)}$$

for salt transport. It should be noted that in Equations (3.4.12) and (3.4.13) $Q_I^{o2} = Q_K^o$, thus the continuity $M_I^{o2} = M_I^o$ is preserved.

When the direct contribution of energy or salt from the overland regime to a junction node *L* (Fig. 3.4-1) is significant, Equations (3.1.121) and (3.1.122) or Equations (3.1.156) and (3.1.157) must be modified

$$\frac{d\rho_W C_W \mathcal{V}_L E_L}{dt} = \sum_i \Phi_{iL}^i + \sum_{O \in N_O} M_O^o \quad or \quad \sum_i \Phi_{iL}^i + \sum_{O \in N_O} M_O^o = 0 \tag{3.4.16}$$

with E_L denoting T_L (where T_L is the temperature at the junction L) for thermal transport or

$$\frac{d\mathcal{V}_L S_L}{dt} = \sum_i \Psi_{iL}^i + \sum_{O \in N_O} M_O^o \quad or \quad \sum_i \Psi_{iL}^i + \sum_{O \in N_O} M_O^o = 0 \tag{3.4.17}$$

with E_L denoting S_L (where S_L is the salinity at the junction L) for salt transport. Additional N_O unknowns have been introduced in Equation (3.4.16) or (3.4.17). For each overland-junction interface node, say O (the right frame in Fig. 3.4-1), the finite element equation written out of Eq. (3.4.11) is

$$C_{O1}^o E_1^o + C_{O2}^o E_2^o + .. + C_{OO}^o E_O^o + .. + C_{OM}^o E_M^o = R_O^o - M_O^o \tag{3.4.18}$$

It is seen that Equation (3.4.18) involves two unknowns, E_O^o and M_O^o. One equation must be supplemented to the finite element equation to close the system. This equation is obtained by formulating energy or salt rates

$$M_O^o = \rho_W C_W Q_O^o \frac{1}{2}\left(\left(1 + sign\left(Q_O^o\right)\right) E_O^o + \left(1 - sign\left(Q_O^o\right)\right) E_L\right) \tag{3.4.19}$$

for thermal transport or

$$M_O^o = Q_O^o \frac{1}{2}\left(\left(1 + sign\left(Q_O^o\right)\right) E_O^o + \left(1 - sign\left(Q_O^o\right)\right) E_L\right) \tag{3.4.20}$$

for salt transport. Finally, the formulation of Φ_{iL}^i or Ψ_{iL}^i is identical to that of M_O^o in Equation (3.4.19) or (3.4.20).

3.4.2 Coupling between 2-D Overland and 3-D Subsurface Flows

The interaction between two-dimensional overland and three-dimensional subsurface flows is rather simple. For every subsurface node (Node *J* in Fig. 3.4-2), there will be associated an overland nodes (Node *I* in Fig. 3.4-2).

3.4.2.1 Couple Flow Rates between the Overland Regime and Subsurface Media.

Numerical approximations of the diffusive water flow equation for two-dimensional overland with finite element methods yield the following matrix

$$
\begin{bmatrix}
-- & -- & -- & -- & -- & -- & -- \\
-- & -- & -- & -- & -- & -- & -- \\
-- & -- & -- & -- & -- & -- & -- \\
A_{I1}^{o} & A_{I2}^{o} & -- & A_{II}^{o} & -- & -- & A_{IN}^{o} \\
-- & -- & -- & -- & -- & -- & -- \\
-- & -- & -- & -- & -- & -- & -- \\
-- & -- & -- & -- & -- & -- & --
\end{bmatrix}
\begin{Bmatrix} H_1^{o} \\ H_2^{o} \\ -- \\ H_I^{o} \\ -- \\ -- \\ H_N^{o} \end{Bmatrix}
=
\begin{Bmatrix} R_1^{o} \\ R_2^{o} \\ -- \\ R_I^{o} \\ -- \\ -- \\ R_N^{o} \end{Bmatrix}
+
\begin{Bmatrix} Q_1^{io} \\ Q_2^{io} \\ -- \\ Q_I^{io} \\ -- \\ -- \\ Q_N^{io} \end{Bmatrix}
\quad \textbf{(3.4.21)}
$$

where the superscript o denotes the overland; A_{IJ} is the I-th row, J-th column of the coefficient matrix [A]; H_I denotes the water surface at Node I; R_I is I-th entry of the load vector {R}; N is the number of nodes in the overland; and Q_I is the rates of water sink/source from/to the overland node I to/from the corresponding subsurface node (e.g., Node J in Fig. 3.4-2) due to infiltration (the superscripts, io, denotes the infiltration from overland). Every overland node I involves two unknowns, H_I^o and and Q_I^{io}. However, Eq. (3.4.21) gives just one algebraic equation for every canal node I. Clearly, one additional algebraic equation is needed every overland node I.

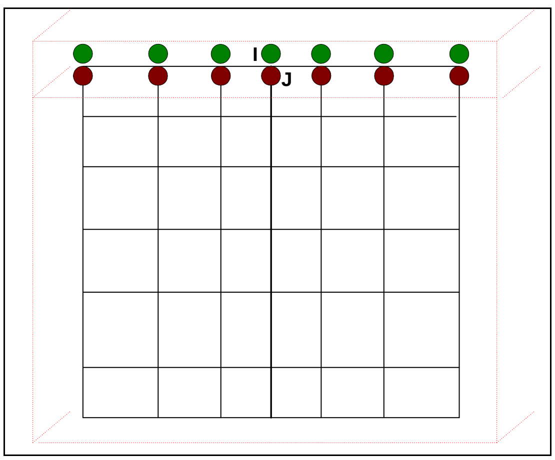

Fig. 3.4-2. Depiction of Interacting Subsurface Nodes and Overland Nodes

Applications of finite element methods to the three-dimensional subsurface flow equation yield the following matrix

$$
\begin{bmatrix}
A_{11}^{s} & A_{12}^{s} & -- & -- & -- & -- & A_{1M}^{s} \\
A_{21}^{s} & -- & -- & -- & -- & -- & A_{2M}^{s} \\
-- & -- & -- & -- & -- & -- & -- \\
A_{J1}^{s} & A_{J2}^{s} & -- & A_{JJ}^{s} & -- & -- & A_{JM}^{s} \\
-- & -- & -- & -- & -- & -- & -- \\
-- & -- & -- & -- & -- & -- & -- \\
A_{M1}^{o} & A_{M2}^{o} & -- & -- & -- & -- & A_{MM}^{o}
\end{bmatrix}
\begin{Bmatrix} H_1^{s} \\ H_2^{s} \\ -- \\ H_J^{s} \\ -- \\ -- \\ H_M^{s} \end{Bmatrix}
=
\begin{Bmatrix} R_1^{s} \\ R_2^{s} \\ -- \\ R_J^{s} \\ -- \\ -- \\ R_M^{s} \end{Bmatrix}
-
\begin{Bmatrix} -- \\ -- \\ -- \\ Q_J^{s} \\ -- \\ -- \\ -- \end{Bmatrix}
\quad \textbf{(3.4.22)}
$$

where the superscript so denotes the subsurface media; A_{IJ} is the I-th row, J-th column of the coefficient matrix [A]; H_J denotes the total head at Node J; R_J is J-th entry of the load vector {R}; M

is the number of nodes in the subsurface media; and Q_J is the rates of water source/sink from/to the overland to/from the subsurface media at node J. Equation (3.4.22) indicates that there is one unknown corresponding to one algebraic equation for every interior node. However, for every algebraic equation corresponding to a subsurface-overland interface node, there are two unknowns, the total head and the flow rate. Therefore, for every subsurface media node interfacing with an overland node, one additional equation is needed. Since for every overland node, there is associated one subsurface-interface node, two additional equations are needed for every overland node I for the two additional unknowns Q_I^{io} and Q_J^s.

The additional equations are obtained by the interface boundary condition between the overland node I and the subsurface media node J as

$$Q_J^s = Q_I^{io}; \qquad H_J^s = H_I^o \quad or \quad Q_I^{io} = K\left(H_J^s - H_I^o\right) \tag{3.4.23}$$

where K is the exchange coefficient representing the property of the medium separating the overland and subsurface media, but not being included as part of the media.

3.4.2.2 Couple thermal or Salt Rate between the Overland Regime and Subsurface Media.

Numerical approximations of thermal or salt transport equation for two-dimensional overland regime with finite element methods yield the following matrix

$$\begin{bmatrix} -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ C_{I1}^o & C_{I2}^o & -- & C_{II}^o & -- & -- & C_{IN}^o \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \end{bmatrix} \begin{Bmatrix} E_1^o \\ E_2^o \\ -- \\ E_I^o \\ -- \\ -- \\ E_N^o \end{Bmatrix} = \begin{Bmatrix} R_1^o \\ R_2^o \\ -- \\ R_I^o \\ -- \\ -- \\ R_N^o \end{Bmatrix} + \begin{Bmatrix} M_1^{io} \\ M_2^{io} \\ -- \\ M_I^{io} \\ -- \\ -- \\ M_N^{io} \end{Bmatrix} \tag{3.4.24}$$

where the superscript o denotes the overland; C_{IJ} is the I-th row, J-th column of the coefficient matrix [C]; E_I denotes the temperature or salinity at Node I; R_I is I-th entry of the load vector {R}; N is the number of nodes in the overland; and M_I is the rate of energy or salt source/sink from/to the subsurface to/from the overland node I (the superscript, io, denotes the infiltration from overland). Every overland node I involves two unknowns, E_I^o, and M_I^{io}. However, Eq. (3.4.24) gives just one algebraic equation for every canal node I. Clearly, one additional algebraic equation is need for every overland node I.

Applications of finite element methods to three-dimensional thermal or salt transport equations for subsurface media yield the following matrix

$$
\begin{bmatrix}
C_{11}^s & C_{12}^s & -- & -- & -- & -- & C_{1M}^s \\
C_{21}^s & -- & -- & -- & -- & -- & C_{2M}^s \\
-- & -- & -- & -- & -- & -- & -- \\
C_{J1}^s & C_{J2}^s & -- & C_{JJ}^s & -- & -- & C_{IM}^s \\
-- & -- & -- & -- & -- & -- & -- \\
-- & -- & -- & -- & -- & -- & -- \\
C_{M1}^s & C_{M2}^s & -- & -- & -- & -- & C_{MM}^s
\end{bmatrix}
\begin{Bmatrix} E_1^s \\ E_2^s \\ -- \\ E_J^s \\ -- \\ -- \\ E_M^s \end{Bmatrix}
=
\begin{Bmatrix} R_1^s \\ R_2^s \\ -- \\ R_J^s \\ -- \\ -- \\ R_M^s \end{Bmatrix}
-
\begin{Bmatrix} -- \\ -- \\ -- \\ M_J^s \\ -- \\ -- \\ -- \end{Bmatrix}
\qquad \textbf{(3.4.25)}
$$

where the superscript s denotes the subsurface media; C_{IJ} is the I-th row, J-th column of the coefficient matrix $[C]$; E_J denotes the temperature or salinity at Node J; R_J is J-th entry of the load vector $\{R\}$; M is the number of nodes in the overland ; and M_J is the rate of thermal or salt sink/source from/to the subsurface node J to/from the corresponding overland node I. Equation (3.4.25) indicates that there is one unknown corresponding to one algebraic equation for every interior node. However, for every algebraic equation corresponding an subsurface-overland interface node, there are two unknowns, the temperature or salinity and the thermal or salt flux. Therefore, for every subsurface-overland interface node, one additional equation is needed. Since for every overland node, there is associated one subsurface-interface nodes, two additional equations are needed for every overland node I and its corresponding subsurface node J for the two additional unknowns M_I^{io} and M_J^s.

The additional equations are obtained from the interface condition between the overland I and the subsurface J as

$$
\begin{aligned}
M_I^{io} &= \rho_W C_W Q_I^{io} \frac{1}{2}\left(\left(1+sign\left(Q_I^{io}\right)\right)E_J^s + \left(1-sign\left(Q_I^{io}\right)\right)E_I^o\right) \quad and \\
M_J^s &= \rho_W C_W Q_J^s \frac{1}{2}\left(\left(1+sign\left(Q_J^s\right)\right)E_J^s + \left(1-sign\left(Q_J^s\right)\right)E_I^o\right)
\end{aligned}
\qquad \textbf{(3.4.26)}
$$

for thermal transport or

$$
\begin{aligned}
M_I^{io} &= Q_I^{io} \frac{1}{2}\left(\left(1+sign\left(Q_I^{io}\right)\right)E_J^s + \left(1-sign\left(Q_I^{io}\right)\right)E_I^o\right) \quad and \\
M_J^s &= Q_J^s \frac{1}{2}\left(\left(1+sign\left(Q_J^s\right)\right)E_J^s + \left(1-sign\left(Q_J^s\right)\right)E_I^o\right)
\end{aligned}
\qquad \textbf{(3.4.27)}
$$

for salt transport. It should be noted that in Equations (3.4.26) or (3.4.27) $Q_I^{io} = Q_J^s$, thus the continuity $M_I^{io} = M_J^s$ is preserved.

3.4.3 Coupling between 3-D Subsurface and 1-D Surface Flows

The interaction between three-dimensional subsurface and one-dimensional river flows involves three options: (1) river is discretized as finite-width and finite-depth on the three-dimensional subsurface media (Fig. 3.4-3), (2) river is discretized as finite-width and zero-depth on the three-dimensional subsurface media (Fig. 3.4-4), and (3) river is discretized as zero-width and zero-depth on the three-dimensional subsurface media (Fig. 3.4-5). Option 1 is the most realistic one.

However, because of the computational demands, it is normally used in small scale studies involving the investigations of infiltration and discharge between river and subsurface media on a local scale. Option 2 is normally used in medium scale studies while Option 3 is normally employed in large scale investigations. In Option 1, for every river node there are associated with a number of subsurface interfacing nodes such as *K*, .., *J*, .., and *L*(Fig. 3.4-3). In Option 2, for every river node there are associated with three subsurface interfacing nodes *K*, *J*, and *L* (Fig. 3.4-4). In Option 3, for every river node there is associated with one subsurface interfacing node *J* (Fig. 3.4-5).

3.4.3.1 Couple Flow Rates between the River Network and the Subsurface Media.

Numerical approximations of the diffusive water flow equation for one-dimensional river with finite element methods yield the following matrix

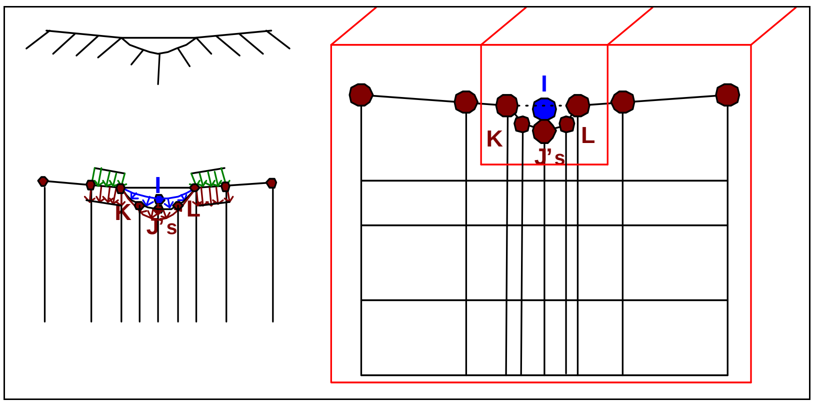

Fig. 3.4-3. Rivers Are Discretized as Finite-Width and Finite-Depth on the Subsurface Media

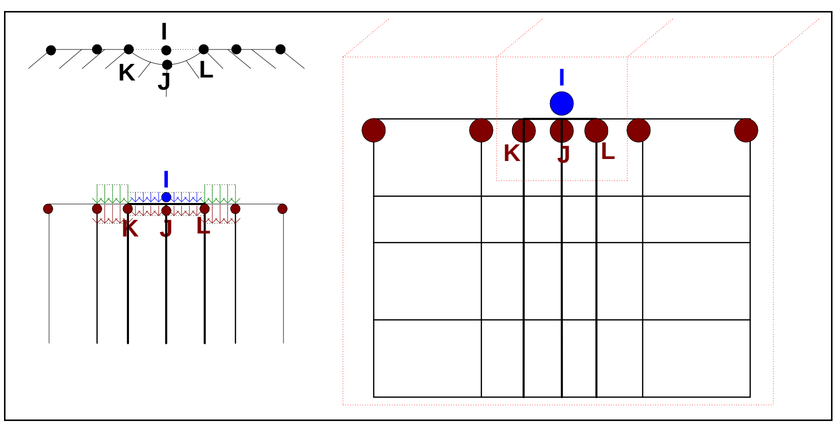

Fig. 3.4-4. Rivers Are Discretized as Finite-Width and Zero-Depth on the Subsurface Media

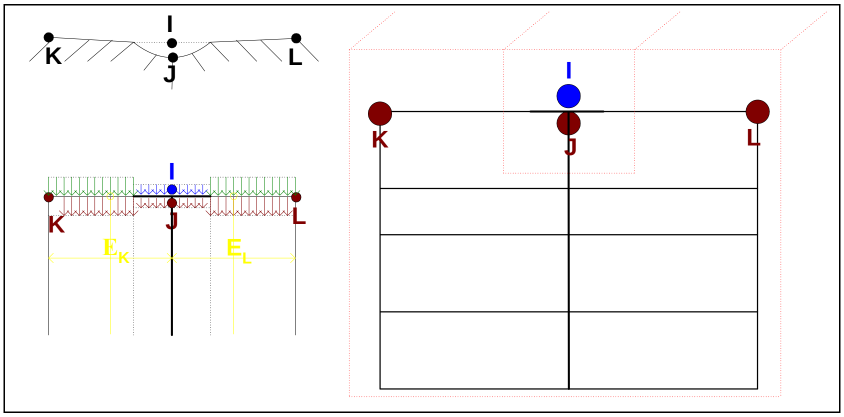

Fig. 3.4-5. Rivers Are Discretized as Zero-Width and Zero-Depth on the Subsurface Media

$$\begin{bmatrix} -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ A^c_{I1} & A^c_{I2} & -- & A^c_{II} & -- & -- & A^c_{IN} \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \end{bmatrix} \begin{Bmatrix} H_1^c \\ H_2^c \\ -- \\ H_I^c \\ -- \\ -- \\ H_N^c \end{Bmatrix} = \begin{Bmatrix} R_1^c \\ R_2^c \\ -- \\ R_I^c \\ -- \\ -- \\ R_N^c \end{Bmatrix} + \begin{Bmatrix} Q_1^{ic} \\ Q_2^{ic} \\ -- \\ Q_I^{ic} \\ -- \\ -- \\ Q_N^{ic} \end{Bmatrix} \quad \textbf{(3.4.28)}$$

where the superscript c denotes the canal (channel, river, or stream); A_{IJ} is the I-th row, J-th column of the coefficient matrix $[A]$; H_I denotes the water surface at Node I; R_I is I-th entry of the load vector $\{R\}$; N is the number of nodes in the canal; Q_I is the rates of water sink/source from/to the river node I to/from the subsurface media. Every canal node I involves two unknowns, H_I^c and Q_I^{ic}. However, Eq. (3.4.28) gives just one algebraic equation for every canal node I. Clearly, one additional algebraic equation is need for every canal node I.

For example, taking Option 2 where there are three nodes associated with one canal node, the applications of finite element methods to three-dimensional subsurface flow equations yield

$$\begin{bmatrix} A^s_{11} & A^s_{12} & -- & -- & -- & -- & A^s_{1M} \\ A^s_{21} & -- & -- & -- & -- & -- & A^s_{2M} \\ -- & -- & -- & -- & -- & -- & -- \\ A^s_{K1} & A^s_{K2} & A^s_{KK} & -- & -- & -- & A^s_{KM} \\ A^s_{J1} & A^s_{J2} & -- & A^s_{JJ} & -- & -- & A^s_{JM} \\ A^s_{L1} & A^s_{L2} & A^s_{>>} & -- & -- & -- & A^s_{LM} \\ -- & -- & -- & -- & -- & -- & -- \\ A^s_{M1} & A^s_{M2} & -- & -- & -- & -- & A^s_{MM} \end{bmatrix} \begin{Bmatrix} H_1^s \\ H_2^s \\ -- \\ H_K^s \\ H_J^s \\ H_L^s \\ -- \\ H_M^s \end{Bmatrix} = \begin{Bmatrix} R_1^s \\ R_2^s \\ -- \\ R_K^s \\ R_J^s \\ R_L^s \\ -- \\ R_M^s \end{Bmatrix} - \begin{Bmatrix} -- \\ -- \\ -- \\ Q_K^s \\ Q_J^s \\ Q_L^s \\ -- \\ -- \end{Bmatrix} \quad \textbf{(3.4.29)}$$

where the superscript s denotes the subsurface meida; A_{IJ} is the I-th row, J-th column of the coefficient matrix $[A]$; H_J denotes the total head at Node J; R_J is J-th entry of the load vector $\{R\}$; M is the number of nodes in the subsurface media; and Q_J is the rate of water source/sink from/to the canal to/from the subsurface via node J. Equation (3.4.29) indicates that there is one unknown corresponding to one algebraic equation for every interior node. However, for every algebraic equation corresponding to a subsurface-canal interface node, there are two unknowns, the total head and the flow rate. Therefore, for every subsurface-river interface node, one additional equation is needed. Since for every canal node, there are associated three subsurface-interface nodes, four additional equations are needed for every canal node I for the four additional unknowns Q_I^{ic}, Q_K^s, Q_J^s, and Q_L^s.

The additional equations are obtained the interface condition between the canal node I and the subsurface nodes K, J, and L as

$$\begin{aligned} &Q_I^{ic} + Q_K^{rain} + Q_L^{rain} = Q_K^s + Q_J^s + Q_L^s; \qquad H_J^s = H_I^c \quad or \quad Q_J^s = K_e\left(H_J^s - H_I^c\right); \\ &H_K^s = H_K^{ponding} \quad or \quad Q_K^s = Q_K^{rain} + \frac{1}{4}Q_I^{ic}; \qquad H_L^s = H_L^{ponding} \quad or \quad Q_L^s = Q_L^{rain} + \frac{1}{4}Q_I^{ic} \end{aligned} \tag{3.4.30}$$

where Q_K^{rain} and Q_L^{rain} are the rainfall fluxes through nodes K and L, respectively; $H_K^{ponding}$ and $H_L^{ponding}$ are the allowable ponding depth at nodes K and L, respectively; and K_e is the exchange coefficient representing the material property of a layer separating the river and subsurface media but the layer is not included in the geometrical discretization.

In Option 1, for every canal node I, there are associated a number of subsurface-interface nodes, say N_S, $(N_S + 1)$ additional equations are needed for every canal node I for the additional unknowns Q_I^{ic}, Q_K^s, .., Q_J^s, .., and Q_L^s. These equations are listed below:

$$\begin{aligned} &Q_I^{ic} + Q_K^{rain} + Q_L^{rain} = Q_K^s + \sum_J^{N_S} Q_J^s + Q_L^s; \\ &H_J^s = H_I^c \quad or \quad Q_J^s = K_e\left(H_J^s - H_I^c\right) \; for\ J \in\ on\ River\ Bottom; \\ &H_K^s = H_K^{ponding} \quad or \quad Q_K^s = Q_K^{rain} + \frac{1}{4}Q_I^{ic}; \qquad H_L^s = H_L^{ponding} \quad or \quad Q_L^s = Q_L^{rain} + \frac{1}{4}Q_I^{ic} \end{aligned} \tag{3.4.31}$$

In Option 3, for every canal node I, there are associated three subsurface-interface nodes K, J, and L as in Option 2. However, while in Option 2, nodes K and J are located at the interactions of river banks and subsurface media, in Option 3, nodes K and L can be located far way from the river banks and node J interacts directly with the canal node I. The four interaction equations are modified according to the continuity of fluxes as

$$\begin{aligned} &Q_J^s = Q_I^{ic} + Q_K^{rain}\left(1 - \frac{P}{E_K}\right) + Q_L^{rain}\left(1 - \frac{P}{E_L}\right); \qquad H_I^c = H_J^s \quad or \quad Q_I^{ic} = K\left(H_J^s - H_I^c\right); \\ &H_K^s = H_K^{ponding} \quad or \quad Q_K^s = Q_K^{rain}; \qquad H_L^s = H_L^{ponding} \quad or \quad Q_L^s = Q_L^{rain} \end{aligned} \tag{3.4.32}$$

where P is the wet perimeter of the canal and E_K and E_L are the element length of KJ and JL, respectively.

3.4.3.2 Couple thermal or Salt Rate between the River Network and the Subsurface.

Numerical approximations of thermal or salt transport equation for one-dimensional river with finite element methods yield the following matrix

$$\begin{bmatrix} -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ C_{I1}^{c} & C_{I2}^{c} & -- & C_{II}^{c} & -- & -- & C_{IN}^{c} \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \end{bmatrix} \begin{Bmatrix} E_1^c \\ E_2^c \\ -- \\ E_I^c \\ -- \\ -- \\ E_N^c \end{Bmatrix} = \begin{Bmatrix} R_1^c \\ R_2^c \\ -- \\ R_I^c \\ -- \\ -- \\ R_N^c \end{Bmatrix} + \begin{Bmatrix} M_1^{ic} \\ M_2^{ic} \\ -- \\ M_I^{ic} \\ -- \\ -- \\ M_N^{ic} \end{Bmatrix} \qquad \textbf{(3.4.33)}$$

where the superscript c denotes the canal (channel, river, or stream); C_{IJ} is the I-th row, J-th column of the coefficient matrix [C]; E_I denotes the temperature or salinity at Node I; R_I is I-th entry of the load vector {R}; N is the number of nodes in the canal; and M_I^{ic} is the rate of energy or salt source/sink from/to the subsurface to/from canal node I due to infiltration/exfiltration. Every canal node I involves two unknowns, E_I^c and M_I^{ic}. However, Eq. (3.4.33) gives just one algebraic equation for every canal node I. Clearly, one additional algebraic equation is need for every canal node I.

For example, taking Option 2 where there are three nodes associated with one canal node, the applications of finite element methods to three-dimensional thermal or salt transport equation in the subsurface media yields

$$\begin{bmatrix} C_{11}^{s} & C_{12}^{s} & -- & -- & -- & -- & C_{1M}^{s} \\ C_{21}^{s} & -- & -- & -- & -- & -- & C_{2M}^{s} \\ -- & -- & -- & -- & -- & -- & -- \\ C_{K1}^{s} & C_{K2}^{s} & C_{KK}^{s} & -- & -- & -- & C_{KM}^{s} \\ C_{J1}^{s} & C_{J2}^{s} & -- & C_{JJ}^{s} & -- & -- & C_{1M}^{s} \\ C_{L1}^{s} & C_{L2}^{s} & C_{<<}^{s} & -- & -- & -- & C_{LM}^{s} \\ -- & -- & -- & -- & -- & -- & -- \\ C_{M1}^{s} & C_{M2}^{s} & -- & -- & -- & -- & C_{MM}^{s} \end{bmatrix} \begin{Bmatrix} E_1^s \\ E_2^s \\ -- \\ E_K^s \\ E_J^s \\ E_L^s \\ -- \\ E_M^s \end{Bmatrix} = \begin{Bmatrix} R_1^s \\ R_2^s \\ -- \\ R_K^s \\ R_J^s \\ R_L^s \\ -- \\ R_M^s \end{Bmatrix} - \begin{Bmatrix} -- \\ -- \\ -- \\ M_K^s \\ M_J^s \\ M_L^s \\ -- \\ -- \end{Bmatrix} \qquad \textbf{(3.4.34)}$$

where the superscript s denotes the subsurface media; C_{IJ} is the I-th row, J-th column of the coefficient matrix [C]; E_J denotes the temperature or salinity at Node J; R_J is J-th entry of the load vector {R}; M is the number of nodes in the overland ; and M_K, M_J and M_L are the rates of thermal or salt sink/source from/to the subsurface water to/from the canal via nodes K, J and L, respectively. Equation (3.4.34) indicates that there is one unknown corresponding to one algebraic equation for every interior node. However, for every algebraic equation corresponding an subsurface-canal interface node, there are two unknowns, the temperature or salinity and the thermal or salt flux. Therefore, for every subsurface-river interface node, one additional equation is needed. Since for every canal node, there are associated three subsurface-interface nodes, four additional equations are needed for every canal node *I* for the four additional unknowns M_I^{ic}, M_K^s, M_J^s, and M_L^s.

These four additional equations are obtained by the interface condition between the canal node *I* and the subsurface nodes *K*, *J*, and *L* as

$$\begin{aligned} M_I^{ic} = Q_I^{ic}\frac{\rho_W C_W}{2}\left(1-sign(Q_I^{ic})\right)E_I^c + \frac{\rho_W C_W}{2}\left(1+sign(Q_I^{ic})\right)\times \\ \left(Q_K^s E_K^s + Q_J^s E_J^s + Q_L^s E_L^s - Q_K^{rains} E_K^{rain} - Q_L^{rains} E_L^{rain}\right) \end{aligned} \qquad \textbf{(3.4.35)}$$

and

$$\begin{aligned} M_K^s &= \rho_W C_W Q_K^s \frac{1}{2}\left(\left(1+sign\left(Q_K^s\right)\right)E_K^s + \left(1-sign\left(Q_K^s\right)\right)E_I^c\right), \\ M_J^s &= \rho_W C_W Q_J^s \frac{1}{2}\left(\left(1+sign\left(Q_J^s\right)\right)E_J^s + \left(1-sign\left(Q_J^s\right)\right)E_I^c\right), \\ M_L^s &= \rho_W C_W Q_L^s \frac{1}{2}\left(\left(1+sign\left(Q_L^s\right)\right)E_L^s + \left(1-sign\left(Q_L^s\right)\right)E_I^c\right) \end{aligned} \qquad \textbf{(3.4.36)}$$

for thermal transport or

$$\begin{aligned} M_I^{ic} = Q_I^{ic}\frac{1}{2}\left(1-sign(Q_I^{ic})\right)E_I^c + \frac{1}{2}\left(1+sign(Q_I^{ic})\right)\times \\ \left(Q_K^s E_K^s + Q_J^s E_J^s + Q_L^s E_L^s - Q_K^{rains} E_K^{rain} - Q_L^{rains} E_L^{rain}\right) \end{aligned} \qquad \textbf{(3.4.37)}$$

and

$$\begin{aligned} M_K^s &= Q_K^s \frac{1}{2}\left(\left(1+sign\left(Q_K^s\right)\right)E_K^s + \left(1-sign\left(Q_K^s\right)\right)E_I^c\right), \\ M_J^s &= Q_J^s \frac{1}{2}\left(\left(1+sign\left(Q_J^s\right)\right)E_J^s + \left(1-sign\left(Q_J^s\right)\right)E_I^c\right), \\ M_L^s &= Q_L^s \frac{1}{2}\left(\left(1+sign\left(Q_L^s\right)\right)E_L^s + \left(1-sign\left(Q_L^s\right)\right)E_I^c\right) \end{aligned} \qquad \textbf{(3.4.38)}$$

for salt transport. For Option 1 and Option 3, the coupling can be done similarly.

3.4.4 Coupling Among River, Overland, and Subsurface Flows

The interaction among one-dimensional river, two-dimensional overland, and three-dimensional subsurface flows involves three options: (1) river is discretized as finite-width and finite-depth on the three-dimensional subsurface media (Fig. 3.4-6), (2) river is discretized as finite-width and zero-depth on the three-dimensional subsurface media (Fig. 3.4-7), and (3) river is discretized as zero-width and zero-depth on the three-dimensional subsurface media (Fig. 3.4-8). Option 1 is the most realistic one. However, because of the computational demands, it is normally used in small scale studies involving the investigations of infiltration and discharge between river and subsurface media on a local scale. Option 2 is normally used in medium scale studies while Option 3 is normally employed in large scale investigations. In Option 1, for every river node there are associated with two overland nodes *M* and *N* and a number of subsurface interfacing nodes such as *K*. , *J*, .., and *L* (Fig. 3.4-6). In Option 2, for every river node *I*, there are associated with two overland nodes *M* and *N* and three subsurface interfacing nodes *K*, *J*, and *L* (Fig. 3.4-7). In Option 3, for every river node *I*, there is associated with two overland nodes *M* and *N* one subsurface node *J* (Fig. 3.4-8).

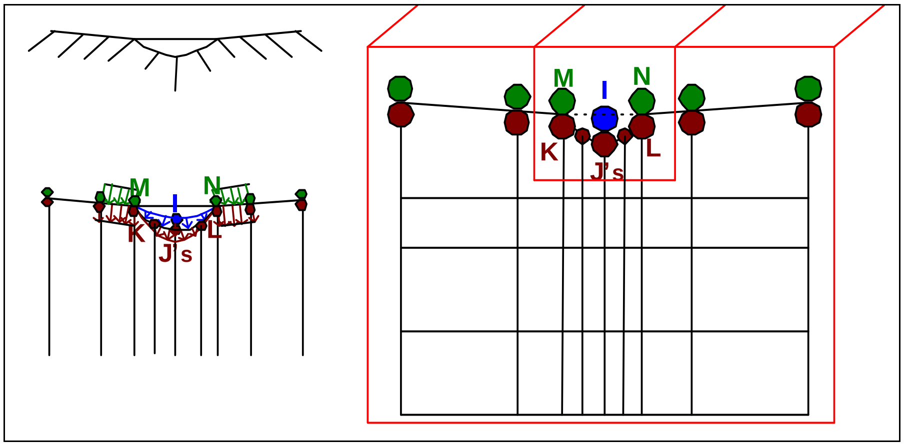

Fig. 3.4-6. Interfacing Nodes for Every River Node when Rivers Are Discretized as Finite-Width and Finite-Depth

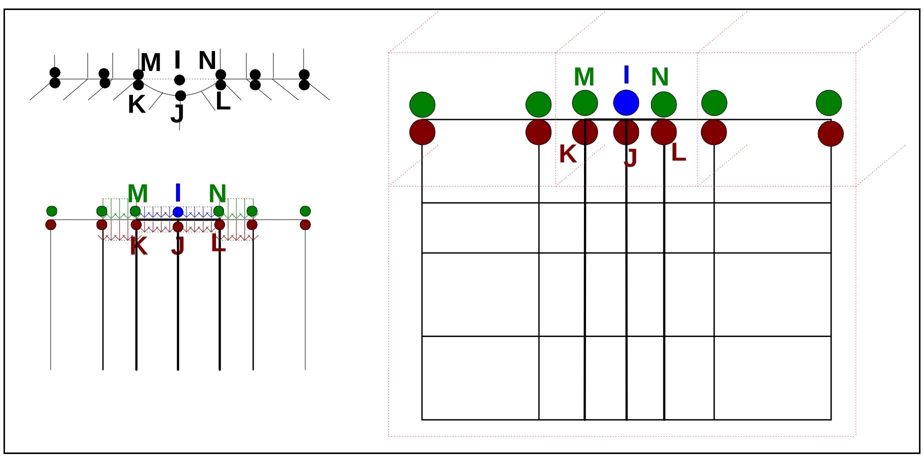

Fig. 3.4-7. Interfacing Nodes for Every River Node when Rivers Are Discretized as Finite-Width and Zero-Depth

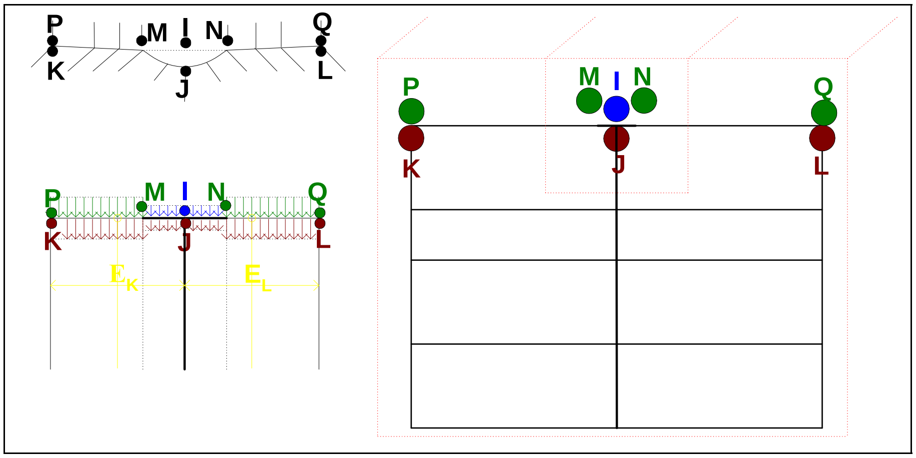

Fig. 3.4-8. Interfacing Nodes for Every River Node when Rivers Are Discretized as Zero-Width and Zero-Depth

3.4.4.1 Couple Flow Rates among River, Overland, and Subsurface Media.

Numerical approximations of flow equations in river, overland, and subsurface would result in a system of algebraic equations. For every river node *I* (Fig. 3.4-7), one or two algebraic equations (for diffusive wave or fully dynamic wave approaches) are obtained governing the water surface (diffusive wave approach) or the water surface and discharge (dynamic wave approach) for the node. The algebraic equation(s) also includes three additional unknowns: two are flow rates from overland to the river via two river banks (Q_I^{o1} and Q_I^{o2})and the other is the flow rate from the subsurface media to river via infiltration/exfiltration ($Q_I^{ic.}$) In the meantime, for the overland node *M* that interfaces with the river node *I* and other subsurface nodes (Fig. 3.4-7), there are two additional unknowns besides the state variables: one is the boundary flux from the overland to the river (Q_M^o) and the other is the infiltration and/or exfiltration flux from overland to the subsurface (Q_M^{io}). Similarly for the overland node *N* that interfaces with the river node *I* and other subsurface nodes (Fig. 3.4-7), there are two additional unknowns besides the state variables: one is the boundary flux from the overland to the river (Q_N^o) and the other is the infiltration and/or exfiltration flux from overland to the subsurface (Q_N^{io}). For the subsurface node *K* that interfaces with the river node *I* and overland node *M* (Fig. 3.4-7), there is one additional unknown (Q_K^s) beside the state variable. Similarly, for the subsurface nodes *L* that interfaces with the river node *I* and overland node *N*, there is one additional unknown (Q_L^s). Finally for the subsurface node J that interfaces with the river node I, there is one additional unknown (Q_J^s) beside the state variable (the pressure head or total head at node *J*). Thus, in Option 2, one needs to set up 10 equations that describe the interactions among flows in river, overland, and subsurface. These ten equations can be derived based on the continuity of fluxes and state variables and formulation of each flux at each individual node as follows.

Interaction between Overland Node *M* and Canal Node *I*. Two equations are obtained based on the continuity of flux and state variable or formulation of flux as

$$Q_M^o = Q_I^{o1}; \qquad H_M^o = H_I^c \quad or \quad Q_I^{o1} = f_1\left(H_M^o, H_I^c\right) \tag{3.4.39}$$

Interaction between Overland Node *N* and Canal Node *I*. Two equations are obtained based on the continuity of flux and state variable or formulation of flux as

$$Q_N^o = Q_I^{o2}; \qquad H_N^o = H_I^c \quad or \quad Q_I^{o2} = f_2\left(H_N^o, H_I^c\right) \tag{3.4.40}$$

Interaction between Overland Node *M*, Subsurface Node *K*, and Canal Node *I*. Two equations are obtained based on the continuity of flux and state variable or formulation of flux as

$$Q_K^s = Q_M^{io} + \frac{1}{4}Q_I^{ic}; \qquad H_K^s = H_M^o \quad or \quad Q_M^{io} = K_e\left(H_K^s - H_M^o\right) \tag{3.4.41}$$

Interaction between River Bank Node *N*, Subsurface Node *L*, and Canal Node *I*. Two equations are obtained based on the continuity of flux and state variable or formulation of flux as

$$Q_L^s = Q_N^{io} + \frac{1}{4}Q_I^{ic}; \qquad H_L^s = H_N^o \quad or \quad Q_N^{io} = K_e\left(H_L^s - H_N^o\right) \tag{3.4.42}$$

Interaction between Subsurface Node *J* and Canal Node *I*. Two equations are obtained based on the continuity of flux and state variable or formulation of flux as

$$Q_J^s = \frac{1}{2}Q_I^{ic}; \qquad H_J^s = H_I^c \quad or \quad Q_J^s = K_e\left(H_J^s - H_I^c\right) \tag{3.4.43}$$

3.4.4.2 Couple thermal or Salt Rate among River, Overland, and Subsurface Media.

Similar to the coupling of flows among river, overland, and subsurface media, the coupling of thermal or salinity transport are achieved by imposing the continuity of energy/salt fluxes and formulation of individual node fluxes.

Interaction between Overland Node *M* and Canal Node *I*. Two equations are obtained based on the continuity of fluxes and the formulation of fluxes as

$$\begin{aligned} M_I^{o1} &= \rho_W C_W Q_I^{o1}\frac{1}{2}\left(\left(1+sign\left(Q_I^{o1}\right)\right)E_M^o + \left(1-sign\left((Q_I^{o1}\right)\right)E_I^c\right) \quad and \\ M_M^o &= \rho_W C_W Q_M^o\frac{1}{2}\left(\left(1+sign\left(Q_M^o\right)\right)E_M^o + \left(1-sign\left((Q_M^o\right)\right)E_I^c\right) \end{aligned} \tag{3.4.44}$$

for thermal transport or

$$\begin{aligned} M_I^{o1} &= Q_I^{o1}\frac{1}{2}\left(\left(1+sign\left(Q_I^{o1}\right)\right)E_M^o + \left(1-sign\left((Q_I^{o1}\right)\right)E_I^c\right) \quad and \\ M_M^o &= Q_M^o\frac{1}{2}\left(\left(1+sign\left(Q_M^o\right)\right)E_M^o + \left(1-sign\left((Q_M^o\right)\right)E_I^c\right) \end{aligned} \tag{3.4.45}$$

for salt transport.

Interaction between Overland Node *N* and Canal Node *I*. Two equations are obtained based on the continuity of fluxes and the formulation of fluxes as

$$M_I^{o2} = \rho_W C_W Q_I^{o2} \frac{1}{2}\Big(\big(1+sign\big(Q_I^{o2}\big)\big)E_N^o + \big(1-sign\big((Q_I^{o2}\big)\big)E_I^c\Big) \quad and$$
$$M_N^o = \rho_W C_W Q_N^o \frac{1}{2}\Big(\big(1+sign\big(Q_N^o\big)\big)E_N^o + \big(1-sign\big((Q_N^o\big)\big)E_I^c\Big) \tag{3.4.46}$$

for thermal transport or

$$M_I^{o2} = Q_I^{o2} \frac{1}{2}\Big(\big(1+sign\big(Q_I^{o2}\big)\big)E_N^o + \big(1-sign\big(Q_I^{o2}\big)\big)E_I^c\Big) \quad and$$
$$M_N^o = Q_N^o \frac{1}{2}\Big(\big(1+sign\big(Q_N^o\big)\big)E_N^o + \big(1-sign\big(Q_N^o\big)\big)E_I^c\Big) \tag{3.4.47}$$

Interaction between Overland Node *M*, Subsurface Node *K*, and Canal Node *I*. Two equations are obtained based on the continuity of fluxes and the formulation of fluxes as

$$M_M^{io} = \rho_W C_W \left\{\frac{1}{2}\big(1-sign\big(Q_M^{io}\big)\big)Q_M^{io}E_M^o + \frac{1}{2}\big(1+sign\big(Q_M^{io}\big)\big)\left(Q_K^s E_K^s - \frac{1}{4}Q_I^{ic}E_I^c\right)\right\} \quad and$$
$$M_K^s = \rho_W C_W \left\{\frac{1}{2}\big(1+sign\big(Q_K^s\big)\big)Q_K^s E_K^s + \frac{1}{2}\big(1-sign\big(Q_K^s\big)\big)\left(Q_M^{io}E_M^o + \frac{1}{4}Q_I^{ic}E_I^c\right)\right\} \tag{3.4.48}$$

for thermal transport and

$$M_M^{io} = \left\{\frac{1}{2}\big(1-sign\big(Q_M^{io}\big)\big)Q_M^{io}E_M^o + \frac{1}{2}\big(1+sign\big(Q_M^{io}\big)\big)\left(Q_K^s E_K^s - \frac{1}{4}Q_I^{ic}E_I^c\right)\right\} \quad and$$
$$M_K^s = \left\{\frac{1}{2}\big(1+sign\big(Q_K^s\big)\big)Q_K^s E_K^s + \frac{1}{2}\big(1-sign\big(Q_K^s\big)\big)\left(Q_M^{io}E_M^o + \frac{1}{4}Q_I^{ic}E_I^c\right)\right\} \tag{3.4.49}$$

for salt transport.

Interaction between River Bank Node *N*, Subsurface Node *L*, and Canal Node *I*. Two equations are obtained based on the continuity of fluxes and the formulation of flux as

$$M_N^{io} = \rho_W C_W \left\{\frac{1}{2}\big(1-sign\big(Q_N^{io}\big)\big)Q_N^{io}E_N^o + \frac{1}{2}\big(1+sign\big(Q_N^{io}\big)\big)\left(Q_L^s E_L^s - \frac{1}{4}Q_I^{ic}E_I^c\right)\right\} \quad and$$
$$M_L^s = \rho_W C_W \left\{\frac{1}{2}\big(1+sign\big(Q_L^s\big)\big)Q_L^s E_L^s + \frac{1}{2}\big(1-sign\big(Q_L^s\big)\big)\left(Q_N^{io}E_N^o + \frac{1}{4}Q_I^{ic}E_I^c\right)\right\} \tag{3.4.50}$$

for thermal transport and

$$M_N^{io} = \left\{\frac{1}{2}\big(1-sign\big(Q_N^{io}\big)\big)Q_N^{io}E_N^o + \frac{1}{2}\big(1+sign\big(Q_N^{io}\big)\big)\left(Q_L^s E_L^s - \frac{1}{4}Q_I^{ic}E_I^c\right)\right\} \quad and$$
$$M_L^s = \left\{\frac{1}{2}\big(1+sign\big(Q_L^s\big)\big)Q_L^s E_L^s + \frac{1}{2}\big(1-sign\big(Q_L^s\big)\big)\left(Q_N^{io}E_N^o + \frac{1}{4}Q_I^{ic}E_I^c\right)\right\} \tag{3.4.51}$$

for salt transport.

Interaction between Subsurface Node *J* and Canal Node *I*. Two equations are obtained based on the continuity of fluxes and the formulation of fluxes as

$$M_I^{ic} = \rho_W C_W \left(\frac{1}{2}\left(1 + sign\left(Q_I^{ic}\right)\right) 2 Q_J^s E_J^s + \frac{1}{2}\left(1 - sign\left(Q_I^{ic}\right)\right) Q_I^{ic} E_I^c \right) \quad and$$
$$M_J^s = \rho_W C_W \left(\frac{1}{2}\left(1 + sign\left(Q_J^s\right)\right) Q_J^s E_J^s + \frac{1}{2}\left(1 - sign\left(Q_J^s\right)\right) \frac{1}{2} Q_I^{ic} E_I^c \right) \tag{3.4.52}$$

for thermal transport and

$$M_I^{ic} = \left(\frac{1}{2}\left(1 + sign\left(Q_I^{ic}\right)\right) 2 Q_J^s E_J^s + \frac{1}{2}\left(1 - sign\left(Q_I^{ic}\right)\right) Q_I^{ic} E_I^c \right) \quad and$$
$$M_J^s = \left(\frac{1}{2}\left(1 + sign\left(Q_J^s\right)\right) Q_J^s E_J^s + \frac{1}{2}\left(1 - sign\left(Q_J^s\right)\right) \frac{1}{2} Q_I^{ic} E_I^c \right) \tag{3.4.53}$$

for salt transport.

3.5 Solving One-Dimensional River/Stream/Canal Network Water Quality Transport Equations

In this section, we present the numerical approaches employed to solve the governing equations of reactive chemical transport in 1-D river/stream/canal networks. Ideally, one would like to use a numerical approach that is accurate, efficient, and robust. Depending on the specific problem at hand, different numerical approaches may be more suitable. For research applications, accuracy is a primary requirement, because one does not want to distort physics due to numerical errors. On the other hand, for large field-scale problems, efficiency and robustness are primary concerns as long as accuracy remains within the bounds of uncertainty associated with model parameters. Thus, to provide accuracy for research applications and efficiency and robustness for practical applications, three coupling strategies were investigated to deal with reactive chemistry. They are: (1) a fully-implicit scheme, (2) a mixed predictor-corrector/operator-splitting method, and (3) an operator-splitting method. For each time-step, we first solve the advective-dispersive transport equation with or without reaction terms, kinetic-variable by kinetic-variable. We then solve the reactive chemical system node-by-node to yield concentrations of all species.

Five numerical options are provided to solve the advective-dispersive transport equations: Option 1- application of the Finite Element Method (FEM) to the conservative form of the transport equations, Option 2 - application of the FEM to the advective form of the transport equations, Option 3 - application of the modified Lagrangian-Eulerian (LE) approach to the Largrangian form of the transport equations, Option 4 - LE approach for all interior nodes and downstream boundary nodes with the FEM applied to the conservative form of the transport equations for the upstream flux boundaries, and Option 5 - LE approach for all interior and downstream boundary nodes with the FEM applied to the advective form of the transport equations for upstream flux boundaries.

3.5.1 One-Dimensional Bed Sediment Balance Equation

At n+1-th time step, the continuity equation for 1-D bed sediment transport, equation (2.5.1), is approximated as follows.

$$\frac{P^{n+1}M_n^{\ n+1} - P^n M_n^{\ n}}{\Delta t} = W_1 P^{n+1}\left(D_n^{\ n+1} - R_n^{\ n+1}\right) + W_2 P^n\left(D_n^{\ n} - R_n^{\ n}\right) \qquad \textbf{(3.5.1.1)}$$

where W_1 and W_2 are time weighting factors satisfying

$$W_1 + W_2 = 1,\ \ 0 < W_1 < 1,\ \ and\ \ 0 < W_2 < 1 \qquad \textbf{(3.5.1.2)}$$

So that

$$M_n^{\ n+1} = \left\{P^n M_n^{\ n} + \left[W_1 P^{n+1}\left(D_n^{\ n+1} - R_n^{\ n+1}\right) + W_2 P^n\left(D_n^{\ n} - R_n^{\ n}\right)\right]\Delta t\right\} \Big/ P^{n+1} \qquad \textbf{(3.5.1.3)}$$

If the calculated $M_n^{\ n+1} < 0$, assign $M_n^{\ n+1} = 0$, so that solve equation (3.5.1.3) and get

$$R_n^{\ n+1} = \left\{P^n M_n^{\ n} + \left[W_1 P^{n+1} D_n^{\ n+1} + W_2 P^n\left(D_n^{\ n} - R_n^{\ n}\right)\right]\Delta t\right\} \Big/ W_1 P^{n+1}\Delta t \qquad \textbf{(3.5.1.4)}$$

3.5.2 Application of the Finite Element Method to the Conservative Form of the Sediment Transport Equations to Solve 1-D Suspended Sediment Transport

Recall governing equation for 1-D suspended sediment transport, equation (2.5.10), as following.

$$\frac{\partial(AS_n)}{\partial t} + \frac{\partial(QS_n)}{\partial x} - \frac{\partial}{\partial x}\left(K_x A\frac{\partial S_n}{\partial x}\right) = M_{S_n}^{\ \ as} + M_{S_n}^{\ \ os1} + M_{S_n}^{\ \ os2} + (R_n - D_n)P, \quad \mathrm{n} \in [1, \mathrm{N_s}] \qquad \textbf{(3.5.2.1)}$$

Assign

$$R_{HS} = (R_n - D_n)P \quad and \quad L_{HS} = 0 \qquad \textbf{(3.5.2.2)}$$

where the right hand side term R_{HS} and left hand side term L_{HS} should be continuously calculated as follows.

$$\begin{aligned} &If\ S_s \le 0,\ M_{S_n}^{\ \ as} = S_s * S_n,\ \ and\ \ L_{HS} = L_{HS} - S_s; \\ &Else\ S_s > 0,\ M_{S_n}^{\ \ as} = M_{S_n}^{\ \ as},\ R_{HS} = R_{HS} + M_{S_n}^{\ \ as} \end{aligned} \qquad \textbf{(3.5.2.3)}$$

$$\begin{aligned} &If\ S_1 \le 0,\ M_{S_n}^{\ \ os1} = S_1 * S_n,\ \ and\ \ L_{HS} = L_{HS} - S_1; \\ &Else\ S_1 > 0,\ M_{S_n}^{\ \ os1} = M_{S_n}^{\ \ os1},\ R_{HS} = R_{HS} + M_{S_n}^{\ \ os1} \end{aligned} \qquad \textbf{(3.5.2.4)}$$

$$\begin{aligned} &If\ S_2 \le 0,\ M_{S_n}^{\ \ os2} = S_2 * S_n,\ \ and\ \ L_{HS} = L_{HS} - S_2 \\ &Else\ S_2 > 0,\ M_{S_n}^{\ \ os2} = M_{S_n}^{\ \ os2},\ R_{HS} = R_{HS} + M_{S_n}^{\ \ os2} \end{aligned} \qquad \textbf{(3.5.2.5)}$$

Then equation (3.5.2.1) is simplified as

$$\frac{\partial(AS_n)}{\partial t}+\frac{\partial(QS_n)}{\partial x}-\frac{\partial}{\partial x}\left(K_x A\frac{\partial S_n}{\partial x}\right)+L_{HS}*S_n=R_{HS} \quad \textbf{(3.5.2.6)}$$

Use Galerkin or Petrov-Galerkin FEM for the spatial discretization of transport equations. For Galerkin method, choose weighting function identical to base functions. For Petrov-Galerkin method, apply weighting function one-order higher than the base function to advection term. Integrate Equation (3.5.2.6) in the spatial dimensions over the entire region as follows.

$$\int_{x_1}^{x_N} N_i\left[\frac{\partial(AS_n)}{\partial t}-\frac{\partial}{\partial x}\left(K_x A\frac{\partial S_n}{\partial x}\right)+L_{HS}*S_n\right]dx+\int_{x_1}^{x_N} W_i\frac{\partial(QS_n)}{\partial x}dx=\int_{x_1}^{x_N} N_i R_{HS}dx \quad \textbf{(3.5.2.7)}$$

Integrating by parts, we obtain

$$\begin{aligned}&\int_{x_1}^{x_N} N_i\frac{\partial(AS_n)}{\partial t}dx-\int_{x_1}^{x_N}\frac{dW_i}{dx}QS_n dx+\int_{x_1}^{x_N}\frac{dN_i}{dx}K_x A\frac{\partial S_n}{\partial x}dx+\int_{x_1}^{x_N} N_i L_{HS}*S_n dx\\&=\int_{x_1}^{x_N} N_i R_{HS}dx-W_i QS_n\Big|_{X_1}^{X_N}+N_i K_x A\frac{\partial S_n}{\partial x}\Big|_{X_1}^{X_N}\end{aligned} \quad \textbf{(3.5.2.8)}$$

Approximate solution S_n by a linear combination of the base functions as shown by Equation (3.5.2.9).

$$S_n\approx\hat{S}_n=\sum_{j=1}^{N}S_{nj}(t)N_j(x) \quad \textbf{(3.5.2.9)}$$

Substituting Equation (3.5.2.9) into Equation (3.5.2.8), we obtain

$$\begin{aligned}&\sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N} N_i\left(\frac{\partial A}{\partial t}+L_{HS}\right)N_j dx-\int_{x_1}^{x_N}\frac{dW_i}{dx}QN_j dx+\int_{x_1}^{x_N}\frac{dN_i}{dx}K_x A\frac{dN_j}{dx}dx\right)S_{nj}(t)\right]\\&+\sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N} N_i AN_j dx\right)\frac{\partial S_{nj}(t)}{\partial t}\right]=\int_{x_1}^{x_N} N_i R_{HS}dx-\sum n\left(W_i QS_n-N_i K_x A\frac{\partial S_n}{\partial x}\right)_b\end{aligned} \quad \textbf{(3.5.2.10)}$$

Equation (3.5.2.10) can be written in matrix form as

$$\left([L1]+[L2]+[L3]\right)\{S_n\}+[M]\left\{\frac{\partial S_n}{\partial t}\right\}=\{SS\}+\{B\} \quad \textbf{(3.5.2.11)}$$

The matrices [L1], [L2], [L3], [M] and load vectors {SS}, {B} are given by

$$M_{ij}=\int_{x_1}^{x_N} N_i AN_j dx \quad \textbf{(3.5.2.12)}$$

$$L1_{ij}=\int_{x_1}^{x_N} N_i\left(\frac{\partial A}{\partial t}+L_{HS}\right)N_j dx \quad \textbf{(3.5.2.13)}$$

$$L2_{ij}=-\int_{x_1}^{x_N}\frac{dW_i}{dx}QN_j dx \quad \textbf{(3.5.2.14)}$$

$$L3_{ij} = \int_{x_1}^{x_N} \frac{dN_i}{dx} K_x A \frac{dN_j}{dx} dx \qquad \textbf{(3.5.2.15)}$$

$$SS_i = \int_{x_1}^{x_N} N_i R_{HS} dx \qquad \textbf{(3.5.2.16)}$$

$$B_i = -n\left(W_i QS_n - N_i K_x A \frac{\partial S_n}{\partial x} \right)_b \qquad \textbf{(3.5.2.17)}$$

where all the terms listed above are calculated with the corresponding time weighting value.

At n+1-th time step, equation (3.5.2.11) is transformed as

$$[L]\left\{W_1 S_n + W_2 S_n{}^p\right\} + [M]\left\{\frac{S_n - S_n{}^p}{\Delta t}\right\} = \{SS\} + \{B\} \;\; where \;\; [L] = [L1] + [L2] + [L3] \qquad \textbf{(3.5.2.18)}$$

So that

$$[CMATRX]\left\{S_n{}^{n+1}\right\} = \{RLD\} \qquad \textbf{(3.5.2.19)}$$

where

$$[CMATRX] = \frac{[M]}{\Delta t} + W_1[L] \qquad \textbf{(3.5.2.20)}$$

$$\{RLD\} = \left(\frac{[M]}{\Delta t} - W_2[L]\right)\left\{S_n{}^n\right\} + \{SS\} + \{B\} \qquad \textbf{(3.5.2.21)}$$

The above equations are used to solve the suspended sediment concentration at interior nodes where boundary term {B} is zero.

The equation employed to determine the suspended sediment at junctions can be derived based on the conservation law of material mass and written as follows.

$$\frac{d\mathcal{V}_j (S_n)_j}{dt} = (M_n{}^s)_j + (M_n{}^{os})_j + [(R_n)_j - (D_n)_j] A_{JTj} + \sum_{k=1}^{NJRTH_j} Flux_k \qquad \textbf{(3.5.2.22)}$$

where $\mathcal{V}_j$ is the junction volume, $(S_n)_j$ is the suspended sediment concentration at the junction, $(M_n{}^s)_j$ is artificial source at the junction, $(M_n{}^{os})_j$ is overland source at the junction, $(R_n)_j$ is erosion rate at the junction, $(D_n)_j$ is deposition rate at the junction, A_{JTj} is the bed area of the junction j, $NJTRH_j$ is the number of river/stream reaches connected to the junction, and $Flux_k$ is the material flux contributed from k-th reach to the junction.

$$Flux_k = n_k\left(Q^k S_n{}^k - K_x A \frac{\partial S_n{}^k}{\partial x} \right) \qquad \textbf{(3.5.2.23)}$$

To solve equation (3.5.2.22) at n+1-th time step, assign

$$L_{HS\,j} = \frac{V_j^{n+1}}{\Delta t} \quad \textbf{(3.5.2.24)}$$

$$R_{HS\,j} = \frac{V_j^{n}(S_n)_j^{\,n}}{\Delta t} + W_2 R_{HS\,j}^{\;\;n} + W_1[(R_n)_j^{\,n+1} - (D_n)_j^{\,n+1}]A_{JT\,j}^{\;\;n+1} \quad \textbf{(3.5.2.25)}$$

where

$$R_{HS\,j}^{\;\;n} = (M_n^{\;s})_j^{\;n} + (M_n^{\;os})_j^{\;n} + [(R_n)_j^{\;n} - (D_n)_j^{\;n}]A_{JT\,j}^{\;\;n} \quad \textbf{(3.5.2.26)}$$

Continue the calculation as follows

$$(M_n^{\;s})_j = \begin{cases} (M_n^{\;s})_j, & \text{if } (S_s)_j > 0 \;\; \Rightarrow \;\; R_{HS\,j} = R_{HS\,j} + W_1(M_n^{\;s})_j \\ (S_s)_j * (S_n)_j, & \text{if } (S_s)_j \le 0 \;\; \Rightarrow \;\; L_{HS\,j} = L_{HS\,j} - W_1(S_s)_j \end{cases} \quad \textbf{(3.5.2.27)}$$

$$(M_n^{\;os})_j = \begin{cases} (M_n^{\;os})_j, & \text{if } (S_{os})_j > 0 \;\; \Rightarrow \;\; R_{HS\,j} = R_{HS\,j} + W_1(M_n^{\;os})_j \\ (S_{os})_j * (S_n)_j, & \text{if } (S_{os})_j \le 0 \;\; \Rightarrow \;\; L_{HS\,j} = L_{HS\,j} - W_1(S_{os})_j \end{cases} \quad \textbf{(3.5.2.28)}$$

Finally, the ordinary differential equation, Eq. (3.5.2.22), is reduced the algebraic equation as follows

$$L_{HS\,j}(S_n)_j - \sum_{k=1}^{NJRTH_j} Flux_k = R_{HS\,j} \quad \textbf{(3.5.2.29)}$$

So that at junction *j*

$$L_{HS\,j}(S_n)_j - W_1 \sum_{k=1}^{NJRTH_j} Flux_k^{\;n+1} = R_{HS\,j} + W_2 \sum_{k=1}^{NJRTH_j} Flux_k^{\;n} \quad \textbf{(3.5.2.30)}$$

For a reach node neighboring the junctions, assign

$$\{RLDW\} = \left(\frac{[M]}{\Delta t} - W_2[L]\right)\{S_n^{\;p}\} + \{SS\} \quad \textbf{(3.5.2.31)}$$

Equation (3.5.2.19) is written as

$$[CMATRX]\{S_n\} + \{Flux\} = \{RLDW\} \quad \textbf{(3.5.2.32)}$$

If $nQ > 0$, flow is going from reach to the junction

$$Flux_k = nQ^k S_n^{\;k} \quad \textbf{(3.5.2.33)}$$

If $nQ < 0$, flow is going from junction to the reach,

$$Flux_k = nQ^k (S_n)_j \quad \textbf{(3.5.2.34)}$$

So that equations (3.5.2.30) and (3.5.2.32) become a set of equation of $(S_n)_j$ and $(S_n)^k$.

For boundary node $i = b$, the boundary term {B} should be calculated as follows.

$$B_i = -n\left(W_i QS_n - N_i K_x A\frac{\partial S_n}{\partial x}\right)_b = -n\left(QS_n - K_x A\frac{\partial S_n}{\partial x}\right)_b \quad \textbf{(3.5.2.35)}$$

Dirichlet boundary condition

$$S_n = S_n(x_b, t) \quad \textbf{(3.5.2.36)}$$

Variable boundary condition

When flow is coming in from outside ($nQ < 0$)

$$n\left(QS_n - AK_x\frac{\partial S_n}{\partial x}\right) = nQS_n(x_b, t) \quad \Rightarrow \quad B_i = -nQS_n(x_b, t) \quad \textbf{(3.5.2.37)}$$

When Flow is going out from inside ($nQ > 0$)

$$-nAK_x\frac{\partial S_n}{\partial x} = 0 \quad \Rightarrow \quad B_i = -nQS_n \quad \textbf{(3.5.2.38)}$$

which must be assembled into the matrix for the boundary point.

Cauchy boundary condition

$$n\left(QS_n - AK_x\frac{\partial S_n}{\partial x}\right) = Q_{Sn}(x_b, t) \quad \Rightarrow \quad B_i = -Q_{S_n}(x_b, t) \quad \textbf{(3.5.2.39)}$$

Neumann boundary condition

$$-nAK_x\frac{\partial S_n}{\partial x} = Q_{Sn}(x_b, t) \quad \Rightarrow \quad B_i = -nQS_n - Q_{S_n}(x_b, t) \quad \textbf{(3.5.2.40)}$$

3.5.3 Application of the Finite Element Method to the Advective Form of the Transport Equations to Solve 1-D Suspended Sediment Transport

Recall governing equation for 1-D suspended sediment transport, equation (2.5.10), as following.

$$\frac{\partial(AS_n)}{\partial t} + \frac{\partial(QS_n)}{\partial x} - \frac{\partial}{\partial x}\left(K_x A\frac{\partial S_n}{\partial x}\right) = M_{S_n}{}^{as} + M_{S_n}{}^{os1} + M_{S_n}{}^{os2} + (R_n - D_n)P, \quad n \in [1, N_s] \quad \textbf{(3.5.3.1)}$$

Conversion to advection form of equation (3.5.3.1) is expressed as

$$A\frac{\partial S_n}{\partial t} + Q\frac{\partial S_n}{\partial x} - \frac{\partial}{\partial x}\left(K_x A\frac{\partial S_n}{\partial x}\right) + \left(\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x}\right)S_n = M_{S_n}{}^{as} + M_{S_n}{}^{os1} + M_{S_n}{}^{os2} + (R_n - D_n)P \quad \textbf{(3.5.3.2)}$$

According to governing equation for 1-D flow, equation (2.1.1), assign

$$R_{HS} = (R_n - D_n)P \quad and \quad L_{HS} = S_S + S_R - S_E + S_I + S_1 + S_2 \qquad \textbf{(3.5.3.3)}$$

where the right hand side term *RHS* and left hand side term *LHS* should be continuously calculated in the same way as that in section 3.5.2. Then equation (3.5.3.2) is simplified as

$$\mathrm{A}\frac{\partial \mathrm{S_n}}{\partial \mathrm{t}} + \mathrm{Q}\frac{\partial \mathrm{S_n}}{\partial \mathrm{x}} - \frac{\partial}{\partial \mathrm{x}}\left(\mathrm{K_x A}\frac{\partial \mathrm{S_n}}{\partial \mathrm{x}}\right) + \mathrm{L_{HS}} * \mathrm{S_n} = \mathrm{R_{HS}} \qquad \textbf{(3.5.3.4)}$$

Use Galerkin or Petrov-Galerkin FEM for the spatial discretization of transport equations. Integrate Equation (3.5.3.4) in the spatial dimensions over the entire region as follows.

$$\int_{x_1}^{x_N} N_i \left[A\frac{\partial S_n}{\partial t} - \frac{\partial}{\partial x}\left(K_x A\frac{\partial S_n}{\partial x}\right) + L_{HS} * S_n \right] dx + \int_{x_1}^{x_N} W_i Q\frac{\partial S_n}{\partial x} dx = \int_{x_1}^{x_N} N_i R_{HS} dx \qquad \textbf{(3.5.3.5)}$$

Integrating by parts for the dispersion/diffusion term, we obtain

$$\begin{aligned} &\int_{x_1}^{x_N} N_i A\frac{\partial S_n}{\partial t} dx + \int_{x_1}^{x_N} W_i Q\frac{\partial S_n}{\partial x} dx + \int_{x_1}^{x_N} \frac{dN_i}{dx} K_x A\frac{\partial S_n}{\partial x} dx + \int_{x_1}^{x_N} N_i L_{HS} * S_n dx \\ &= \int_{x_1}^{x_N} N_i R_{HS} dx + N_i K_x A\frac{\partial S_n}{\partial x}\bigg|_{x_1}^{x_N} \end{aligned} \qquad \textbf{(3.5.3.6)}$$

Approximate solution S_n by a linear combination of the base functions as shown by Equation (3.5.3.7).

$$S_n \approx \hat{S}_n = \sum_{j=1}^{N} S_{nj}(t) N_j(x) \qquad \textbf{(3.5.3.7)}$$

Substituting Equation (3.5.3.7) into Equation (3.5.3.6), we obtain

$$\begin{aligned} &\sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N} N_i L_{HS} N_j dx + \int_{x_1}^{x_N} W_i Q\frac{dN_j}{dx} dx + \int_{x_1}^{x_N} \frac{dN_i}{dx} K_x A\frac{dN_j}{dx} dx\right) S_{nj}(t)\right] \\ &+ \sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N} N_i A N_j dx\right)\frac{\partial S_{nj}(t)}{\partial t}\right] = \int_{x_1}^{x_N} N_i R_{HS} dx + \sum n\left(N_i K_x A\frac{\partial S_n}{\partial x}\right)_b \end{aligned} \qquad \textbf{(3.5.3.8)}$$

Equation (3.5.3.8) can be written in matrix form as

$$([L1] + [L2] + [L3])\{S_n\} + [M]\left\{\frac{\partial S_n}{\partial t}\right\} = \{SS\} + \{B\} \qquad \textbf{(3.5.3.9)}$$

The matrices $[L1]$, $[L2]$, $[L3]$, $[M]$ and load vectors $\{SS\}$, $\{B\}$ are given by

$$M_{ij} = \int_{x_1}^{x_N} N_i A N_j dx \qquad \textbf{(3.5.3.10)}$$

$$L1_{ij} = \int_{x_1}^{x_N} N_i L_{HS} N_j dx \qquad \textbf{(3.5.3.11)}$$

$$L2_{ij} = \int_{x_1}^{x_N} W_i Q \frac{dN_j}{dx} dx \quad \textbf{(3.5.3.12)}$$

$$L3_{ij} = \int_{x_1}^{x_N} \frac{dN_i}{dx} K_x A \frac{dN_j}{dx} dx \quad \textbf{(3.5.3.13)}$$

$$SS_i = \int_{x_1}^{x_N} N_i R_{HS} dx \quad \textbf{(3.5.3.14)}$$

$$B_i = -n\left(-N_i K_x A \frac{\partial S_n}{\partial x}\right)_b \quad \textbf{(3.5.3.15)}$$

where all the terms listed above are calculated with the corresponding time weighting value.

At n+1-th time step, equation (3.5.3.9) is approximated as

$$[L]\left\{W_1 S_n^{\,n+1} + W_2 S_n^{\,n}\right\} + [M]\left\{\frac{S_n^{\,n+1} - S_n^{\,n}}{\Delta t}\right\} = \{SS\} + \{B\} \text{ where } [L] = [L1] + [L2] + [L3] \quad \textbf{(3.5.3.16)}$$

So that

$$[CMATRX]\left\{S_n^{\,n+1}\right\} = \{RLD\} \quad \textbf{(3.5.3.17)}$$

where

$$[CMATRX] = \frac{[M]}{\Delta t} + W_1[L] \quad \textbf{(3.5.3.18)}$$

$$\{RLD\} = \left(\frac{[M]}{\Delta t} - W_2[L]\right)\left\{S_n^{\,n}\right\} + \{SS\} + \{B\} \quad \textbf{(3.5.3.19)}$$

The above equations are used to solve the suspended sediment concentration at interior nodes where boundary term $\{B\}$ is zero.

At internal boundary points neighboring the junctions, assign

$$\{RLDW\} = \left(\frac{[M]}{\Delta t} - W_2[L]\right)\left\{S_n^{\,p}\right\} + \{SS\} + \{nQS_n\} \quad \textbf{(3.5.3.20)}$$

Equation (3.5.3.17) is modified as

$$[CMATRX]\{S_n\} + \{Flux\} = \{RLDW\} \quad \textbf{(3.5.3.21)}$$

So that junction concentration can be solved by equations (3.5.2.30) and (3.5.3.21).

For a global boundary node $i = b$, the boundary term $\{B\}$ should be calculated as follows.

$$B_i = n\left(N_i K_x A \frac{\partial S_n}{\partial x}\right)_b = n\left(K_x A \frac{\partial S_n}{\partial x}\right)_b \quad \textbf{(3.5.3.22)}$$

Dirichlet boundary condition

$$S_n = S_n(x_b, t) \qquad \textbf{(3.5.3.23)}$$

Variable boundary condition

When flow is coming in from outside ($nQ < 0$)

$$n\left(QS_n - AK_x\frac{\partial S_n}{\partial x}\right) = nQS_n(x_b,t) \quad \Rightarrow \quad B_i = nQS_n - nQS_n(x_b,t) \qquad \textbf{(3.5.3.24)}$$

When Flow is going out from inside ($nQ > 0$)

$$-nAK_x\frac{\partial S_n}{\partial x} = 0 \quad \Rightarrow \quad B_i = 0 \qquad \textbf{(3.5.3.25)}$$

Cauchy boundary condition

$$\mathrm{n}\left(\mathrm{QS_n} - \mathrm{AK_x}\frac{\partial \mathrm{S_n}}{\partial \mathrm{x}}\right) = \mathrm{Q_{Sn}}(\mathrm{x_b},\mathrm{t}) \Rightarrow \mathrm{B_i} = \mathrm{nQS_n} - \mathrm{Q_{S_n}}(\mathrm{x_b},\mathrm{t}) \qquad \textbf{(3.5.3.26)}$$

Neumann boundary condition

$$-nAK_x\frac{\partial S_n}{\partial x} = Q_{Sn}(t) \quad \Rightarrow \quad B_i = -Q_{S_n}(t) \qquad \textbf{(3.5.3.27)}$$

3.5.4 Application of the Modified Lagrangian-Eulerian Approach to the Largrangian Form of the Transport Equations to Solve 1-D Suspended Sediment Transport

Recall governing equation for 1-D suspended sediment transport in advection form, equation (3.5.3.2), as follows

$$A\frac{\partial S_n}{\partial t} + Q\frac{\partial S_n}{\partial x} - \frac{\partial}{\partial x}\left(K_x A\frac{\partial S_n}{\partial x}\right) + \left(\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x}\right)S_n = M_{s_n}{}^{as} + M_{s_n}{}^{os1} + M_{s_n}{}^{os2} + (R_n - D_n)P \qquad \textbf{(3.5.4.1)}$$

Assign and calculate R_{HS} and L_{HS} the same as that in section (3.5.3). Then equation (3.5.4.1) is simplified as

$$A\frac{\partial S_n}{\partial t} + Q\frac{\partial S_n}{\partial x} - \frac{\partial}{\partial x}\left(K_x A\frac{\partial S_n}{\partial x}\right) + L_{HS} * S_n = R_{HS} \qquad \textbf{(3.5.4.2)}$$

Equation (3.5.4.2) in the Lagrangian and Eulerian form is written as follows. In the Lagrangian step

$$A\frac{dS_n}{d\tau} = A\frac{\partial S_n}{\partial t} + Q\frac{\partial S_n}{\partial x} = 0 \quad \Rightarrow \quad \frac{\partial S_n}{\partial t} + V\frac{\partial S_n}{\partial x} = 0 \qquad \textbf{(3.5.4.3)}$$

where τ is the tracking time, and particle-tracking velocity V is the flow velocity. In the Eulerian step

$$A\frac{dS_n}{d\tau} - \frac{\partial}{\partial x}\left(K_x A\frac{\partial S_n}{\partial x}\right) + L_{HS} * S_n = R_{HS} \qquad \textbf{(3.5.4.4)}$$

Equation (3.5.4.4) written in a slightly different form is shown as follows.

$$\frac{dS_n}{d\tau} - D + K * S_n = R_L \tag{3.5.4.5}$$

where

$$AD = \frac{\partial}{\partial x}\left(K_x A \frac{\partial S_n}{\partial x}\right) \tag{3.5.4.6}$$

$$K = \frac{L_{HS}}{A} \tag{3.5.4.7}$$

$$R_L = \frac{R_{HS}}{A} \tag{3.5.4.8}$$

Integrating Eq. (3.5.4.5) along a characteristic line to yield the following matrix equation as

$$\begin{aligned} &\frac{[\mathbf{U}]}{\Delta\tau}\left\{S_n^{\ n+1}\right\} - W_1\left\{D^{n+1}\right\} + W_1\left[\mathbf{K}^{n+1}\right]\left\{S_n^{\ n+1}\right\} = \\ &\frac{[\mathbf{U}]}{\Delta\tau}\left\{S_n^{\ *}\right\} + W_2\left\{D^*\right\} - W_2\left\{\left(KS_n\right)^*\right\} + W_1\left\{R_L^{\ n+1}\right\} + W_2\left\{R_L^{\ *}\right\} \end{aligned} \tag{3.5.4.9}$$

where * corresponds to the previous time step value at the location where node i is backwardly tracked in the Lagrangian step, [**U**] is the unit matrix, and [$\mathbf{K}^{n+1}$] is a diagonal matrix with *K* calculated at the (*n*+1)-th time step as its diagonal components..

The diffusion term D expressed in term of S_n is solved by the following procedure. Approximate *D* by a linear combination of the base functions as follows.

$$D \approx \hat{D} = \sum_{j=1}^{N} D_j(t) N_j(x) \tag{3.5.4.10}$$

Applying the Galerkin finite element method to Eq. (3.5.4.6), we obtain

$$\int_{x_1}^{x_N} N_i ADdx = \int_{x_1}^{x_N} N_i A \sum_{j=1}^{N} D_j(t) N_j(x) dx = \int_{x_1}^{x_N} N_i \frac{\partial}{\partial x}\left(K_x A \frac{\partial S_n}{\partial x}\right) dx \tag{3.5.4.11}$$

Integrating by parts, we obtain

$$\sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N} N_i AN_j dx\right) * D_j\right] = -\int_{x_1}^{x_N} \frac{dN_i}{dx} K_x A \frac{\partial S_n}{\partial x} dx + N_i K_x A \frac{\partial S_n}{\partial x}\bigg|_{X_1}^{X_N} \tag{3.5.4.12}$$

Approximate S_n by a linear combination of the base functions as follows.

$$S_n \approx \hat{S}_n = \sum_{j=1}^{N} S_{nj}(t) N_j(x) \tag{3.5.4.13}$$

Substituting Eq. (3.5.4.13) into Eq. (3.5.4.12), we have

$$\sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N} N_i A N_j dx\right) * D_j\right] = -\sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N} \frac{dN_i}{dx} K_x A \frac{dN_j}{\partial x} dx\right) * (S_n)_j\right] + N_i K_x A \frac{\partial S_n}{\partial x}\bigg|_{X_1}^{X_N} \tag{3.5.4.14}$$

Assign matrices [QA], [QD] and load vector {B} as following.

$$QA_{ij} = \int_{x_1}^{x_N} N_i A N_j dx \tag{3.5.4.15}$$

$$QD_{ij} = \int_{x_1}^{x_N} \frac{dN_i}{dx} K_x A \frac{dN_j}{dx} dx \tag{3.5.4.16}$$

$$B_i = \left(nN_i K_x A \frac{\partial S_n}{\partial x}\right)_b \tag{3.5.4.17}$$

Equation (3.5.4.14) is expressed as

$$[QA]\{D\} = -[QD]\{S_n\} + \{QB\} \tag{3.5.4.18}$$

Lump matrix [QA] into diagonal matrix and update

$$QD_{ij} = QD_{ij} / QA_{ii} \tag{3.5.4.19}$$

$$B_i = QB_i / QA_{ii} \tag{3.5.4.20}$$

Then

$$\{D\} = -[QD]\{S_n\} + \{B\} \tag{3.5.4.21}$$

where {B} is calculated as follows

Dirichlet boundary condition

$$S_n = S_n(x_b, t) \;\;\Rightarrow\;\; B_i = nN_i K_x A \frac{(S_n)_j - S_n(x_b,t)}{\Delta x} \Big/ QA_{ii} \tag{3.5.4.22}$$

where *j* is the interior node connected to the boundary node.

Variable boundary condition

When flow is coming in from outside (nQ < 0)

$$n\left(QS_n - AK_x \frac{\partial S_n}{\partial x}\right) = nQS_n(x_b,t) \;\;\Rightarrow\;\; B_i = \left[nQS_n - nQS_n(x_b,t)\right] / QA_{ii} \tag{3.5.4.23}$$

When Flow is going out from inside ($nQ > 0$)

$$-nAK_x \frac{\partial S_n}{\partial x} = 0 \;\;\Rightarrow\;\; B_i = 0 \tag{3.5.4.24}$$

Cauchy boundary condition

$$n\left(QS_n - AK_x\frac{\partial S_n}{\partial x}\right) = Q_{Sn}(x_b,t) \quad \Rightarrow \quad B_i = \left[nQS_n - Q_{Sn}(x_b,t)\right]/QA_{ii} \qquad \textbf{(3.5.4.25)}$$

Neumann boundary condition

$$-nAK_x\frac{\partial S_n}{\partial x} = Q_{Sn}(x_b,t) \quad \Rightarrow \quad B_i = -Q_{Sn}(x_b,t)/QA_{ii} \qquad \textbf{(3.5.4.26)}$$

According to equation (3.5.4.21), Equation (3.5.4.9) can be modified as follows

$$[CMATRX]\left\{S_n^{\,n+1}\right\} = \left\{RLD\right\} \qquad \textbf{(3.5.4.27)}$$

where

$$[CMATRX] = \frac{[U]}{\Delta\tau} + W_1[QD^{n+1}] + W_1\left[K^{n+1}\right] \qquad \textbf{(3.5.4.28)}$$

$$\left\{RLD\right\} = \frac{[U]}{\Delta\tau}\left\{S_n^{\,*}\right\} + W_2\left\{D^*\right\} - W_2\left\{\left(KS_n\right)^*\right\} + W_1\left\{R_L^{\,n+1}\right\} + W_2\left\{R_L^{\,*}\right\} + W_1\{B^{n+1}\} \qquad \textbf{(3.5.4.29)}$$

The above equations are used to solve the suspended sediment concentration at interior nodes where boundary term $\{B^{n+1}\}$ is zero.

At the junctions, if $nQ > 0$, flow is going from the reach to the junction, assign

$$\left\{RLDW\right\} = \left\{RLD\right\} + \left\{nQS_n\right\}/QA_{ii}^{\,n+1} - W_1\left\{B^{n+1}\right\} - W_2[QB^n]\{S_n^{\,n}\}/QA_{ii} \qquad \textbf{(3.5.4.30)}$$

Equation (3.5.4.30) is written as

$$[CMATRX]\left\{S_n^{\,n+1}\right\} + \left\{Flux/QA_{ii}^{\,n+1}\right\} = \left\{RLDW\right\} \qquad \textbf{(3.5.4.31)}$$

If $nQ < 0$, flow in going from junction to the reach, apply equation (3.5.2.23)

$$Flux_i = n\left[Q(S_n)_i - K_xA\frac{(S_n)_j - (S_n)_i}{\Delta x}\right] \qquad \textbf{(3.5.4.32)}$$

where j is the interior node connected to the junction node i.

Junction concentration can be solved with equations (3.5.2.30), (3.5.4.31) and (3.5.4.32).

For boundary node $i = b$, the boundary term $\{B^{n+1}\}$ in equation (3.5.4.29) should be calculated as follows.

Dirichlet boundary condition

$$S_n = S_n(x_b,t) \qquad \textbf{(3.5.4.33)}$$

The above equation is used for Dirichlet boundary node rather than equation (3.5.4.29).

Variable boundary condition

When flow is coming in from outside ($nQ < 0$), equation (3.5.4.29) cannot be applied because $\Delta\tau$ equations zero. Applying boundary condition, we have

$$n\left[Q(S_n)_i - AK_x\frac{(S_n)_j-(S_n)_i}{\Delta x}\right] = nQS_n(x_b,t) \quad \textbf{(3.5.4.34)}$$

where j is the interior node connected to the boundary node i.

When Flow is going out from inside ($nQ > 0$), the boundary term $\{B^{n+1}\}$ in equation (3.5.4.29) should be calculated as follows.

$$-nAK_x\frac{\partial S_n}{\partial x} = 0 \quad \Rightarrow \quad B_i = 0 \quad \textbf{(3.5.4.35)}$$

Cauchy boundary condition

Equation (3.5.4.29) cannot be applied because $\Delta\tau$ equations zero. Applying boundary condition, we have

$$n\left[Q(S_n)_i - AK_x\frac{(S_n)_j-(S_n)_i}{\Delta x}\right] = Q_{Sn}(x_b,t) \quad \textbf{(3.5.4.36)}$$

Neumann boundary condition

The boundary term $\{B^{n+1}\}$ in equation (3.5.4.29) should be calculated as follows.

$$-n\mathrm{AK_x}\frac{\partial \mathrm{S_n}}{\partial \mathrm{x}} = \mathrm{Q_{Sn}(x_b,t)} \quad \Rightarrow \quad B_i = -Q_{Sn}(x_b,t)/QA_{ii} \quad \textbf{(3.5.4.37)}$$

3.5.5 Aplication of the Lagrangian-Eulerian Approach for All Interior Nodes and Downstream Boundary Nodes with the Finite Element Method Applied to the Conservative Form of the Transport Equations for the Upstream Flux Boundaries to Solve 1-D Suspended Sediment Transport

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.5.4, and the matrix equation for junction and upstream boundary nodes is obtained through the same procedure as that in section 3.5.2.

3.5.6 Application of the Lagrangian-Eulerian Approach for All Interior Nodes and Downstream Boundary Nodes with the Finite Element Method Applied to the Advective Form of the Transport Equations for the Upstream Flux Boundaries to Solve

1-D Suspended Sediment Transport

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.5.4, and the matrix equation for junction and upstream boundary nodes is obtained through the same procedure as that in section 3.5.3.

3.5.7 Finite Application of the Finite Element Method to the Conservative Form of the Transport Equations to Solve 1-D Kinetic Variable Transport

3.5.7.1 Fully implicit scheme

Recall the continuity equation for kinetic-variables, equation (2.5.44), can be written in slightly different form by expanding the time derivative term as

$$A\frac{\partial E_n}{\partial t}+\frac{\partial A}{\partial t}E_n+\frac{\partial (QE_n{}^m)}{\partial x}-\frac{\partial}{\partial x}\left(K_x A\frac{\partial E_n{}^m}{\partial x}\right)=M_{E_n}{}^{as}+M_{E_n}{}^{rs}+M_{E_n}{}^{is}+M_{E_n}{}^{os1}+M_{E_n}{}^{os2}+AR_{E_n} \quad \textbf{(3.5.7.1.1)}$$

where E_n is the concentration of the n-th kinetic variable, $E_n{}^m$ is the mobile concentration of the n-th kinetic variable, $M_{E_n}{}^{as}$ is the rate of artificial source of the n-th kinetic variable E_n, $M_{E_n}{}^{rs}$ is the rate of rainfall source/evaporation sink of the n-th kinetic variable E_n, , $M_{E_n}{}^{os1}$ is the rate of overland source from Bank *1* of the n-th kinetic variable E_n, $M_{E_n}{}^{os2}$ is the rate of overland source from Bank *2* of the n-th kinetic variable E_n, $M_{E_n}{}^{is}$ is the rate of exfiltration source of the n-th kinetic variable E_n, and R_{E_n} and is the rate of reaction of the n-th kinetic variable E_n.

At ($n+1$)-th time step, equation (3.5.7.1.1) is approximated by

$$A\frac{(E_n)^{n+1}-(E_n)^n}{\Delta t}+\frac{\partial A}{\partial t}E_n+\frac{\partial (QE_n{}^m)}{\partial x}-\frac{\partial}{\partial x}\left(K_x A\frac{\partial E_n{}^m}{\partial x}\right)=M_{E_n}{}^{as}+M_{E_n}{}^{rs}+M_{E_n}{}^{is}+M_{E_n}{}^{os1}+M_{E_n}{}^{os2}+AR_{E_n} \quad \textbf{(3.5.7.1.2)}$$

where the superscripts ^{n} and $^{n+1}$ represent the time step number. Terms without superscript should be the corresponding average values calculated with time weighting factors W_1 and W_2.

According to the fully-implicit scheme, equation (3.5.7.1.2) can be separated into two equations as follows

$$A\frac{(E_n)^{n+1/2}-(E_n)^n}{\Delta t}+\frac{\partial A}{\partial t}E_n+\frac{\partial (QE_n{}^m)}{\partial x}-\frac{\partial}{\partial x}\left(K_x A\frac{\partial E_n{}^m}{\partial x}\right)=M_{E_n}{}^{as}+M_{E_n}{}^{rs}+M_{E_n}{}^{is}+M_{E_n}{}^{os1}+M_{E_n}{}^{os2}+AR_{E_n} \quad \textbf{(3.5.7.1.3)}$$

$$\frac{(E_n)^{n+1}-(E_n)^{n+1/2}}{\Delta t}=0 \quad \textbf{(3.5.7.1.4)}$$

First, we express $E_n{}^m$ in terms of $(E_n{}^m/E_n)\cdot E_n$ to make E_n's as primary dependent variables, so that $E_n{}^{n+1/2}$ can be solved from Eq. (3.5.7.1.3). Second, we solve equation (3.5.7.1.4) together with algebraic equations for equilibrium reactions using BIOGEOCHEM to obtain all individual species concentrations. Iteration between these two steps is needed because the new reaction terms R_n^{n+1} and the equation coefficients in equation (3.5.7.1.3) need to be updated with the calculation results of (3.5.7.1.4). To improve the standard SIA method, the nonlinear reaction terms are approximated by the Newton-Raphson linearization.

To solve equation (3.5.7.1.3), assign

$$R_{HSn} = 0 \quad and \quad L_{HSn} = 0 \qquad \textbf{(3.5.7.1.5)}$$

Then the right hand side R_{HSn} and left hand side L_{HSn} should be continuously calculated as following

$$M_{E_n}^{\ rs} = \begin{cases} S_R * E_{n^{rs}}, \ \ if \ S_R > 0 \ \Rightarrow \ R_{HSn} = R_{HSn} + M_{E_n}^{\ rs} \\ S_R * E_n^{\ m}, \ \ if \ S_R \le 0 \ \Rightarrow \ L_{HSn} = L_{HSn} - S_R \end{cases} \qquad \textbf{(3.5.7.1.6)}$$

$$M_{E_n}^{\ as} = \begin{cases} S_S * E_{n^{as}}, \ \ if \ S_S > 0 \ \Rightarrow \ R_{HSn} = R_{HSn} + M_{E_n}^{\ as}, \\ S_S * E_n^{\ m}, \ \ if \ S_S \le 0 \ \Rightarrow \ L_{HSn} = L_{HSn} - S_S \end{cases} \qquad \textbf{(3.5.7.1.7)}$$

$$M_{E_n}^{\ os1} = \begin{cases} S_1 * E_n^{\ m}{}_{os1}, \ \ if \ S_1 > 0 \ \Rightarrow \ R_{HSn} = R_{HSn} + M_{E_n}^{\ os1} \\ S_1 * E_n^{\ m}, \ \ if \ S_1 \le 0 \ \Rightarrow \ L_{HSn} = L_{HSn} - S_1 \end{cases} \qquad \textbf{(3.5.7.1.8)}$$

$$M_{E_n}^{\ os2} = \begin{cases} S_2 * E_n^{\ m}{}_{os2}, \ \ if \ S_2 > 0 \ \Rightarrow \ R_{HSn} = R_{HSn} + M_{E_n}^{\ os2} \\ S_2 * E_n^{\ m}, \ \ if \ S_2 \le 0 \ \Rightarrow \ L_{HSn} = L_{HSn} - S_2 \end{cases} \qquad \textbf{(3.5.7.1.9)}$$

$$M_{E_n}^{\ is} = \begin{cases} S_I * E_n^{\ m}{}_{is}, \ \ if \ S_I > 0 \ \Rightarrow \ R_{HSn} = R_{HSn} + M_{E_n}^{\ is} \\ S_I * E_n^{\ m}, \ \ if \ S_I \le 0 \ \Rightarrow \ L_{HSn} = L_{HSn} - S_I \end{cases} \qquad \textbf{(3.5.7.1.10)}$$

where $E_{n^{rs}}$ is the concentration of E_n in the rainfall source, $E_{n^{es}}$ is the concentration of E_n in the evaporation source, $E_{n^{as}}$ is the concentration of E_n in the artificial source, $E_{n^{os1}}$ is the concentration of E_n in the overland source from bank 1, $E_{n^{os2}}$ is the concentration of E_n in the overland source from bank 2, and $E_{n^{is}}$ is the concentration of E_n in the exfiltration source from subsurface media.

Equation (3.5.7.1.3) is then simplified as

$$A\frac{(E_n)^{n+1/2} - (E_n)^n}{\Delta t} + \frac{\partial A}{\partial t}E_n + \frac{\partial (QE_n^{\ m})}{\partial x} - \frac{\partial}{\partial x}\left(K_x A \frac{\partial E_n^{\ m}}{\partial x}\right) + L_{HSn} * E_n^{\ m} = R_{HSn} + AR_{E_n} \qquad \textbf{(3.5.7.1.11)}$$

Express E_n^m in terms of $(E_n^m/E_n)E_n$ to make E_n*'s* as primary dependent variables,

$$A\frac{\partial E_n}{\partial t} + \frac{\partial A}{\partial t}E_n + \frac{\partial}{\partial x}\left(Q\frac{E_n^{\ m}}{E_n}E_n\right) - \frac{\partial}{\partial x}\left(K_x A \frac{E_n^{\ m}}{E_n}\frac{\partial E_n}{\partial x}\right) - \frac{\partial}{\partial x}\left(K_x A \frac{\partial (E_n^{\ m}/E_n)}{\partial x}E_n\right) + L_{HSn}\frac{E_n^{\ m}}{E_n} = R_{HSn} + AR_{E_n} \qquad \textbf{(3.5.7.1.12)}$$

Use Galerkin or Petrov-Galerkin FEM for the spatial discretization of transport equations. Integrate Equation (3.5.7.1.12) in the spatial dimensions over the entire region as follows.

$$\int_{x_1}^{x_N} N_i \left[A\frac{\partial E_n}{\partial t} - \frac{\partial}{\partial x}\left(K_x A \frac{E_n{}^m}{E_n}\frac{\partial E_n}{\partial x}\right)\right] dx + \int_{x_1}^{x_N} W_i \left[\frac{\partial}{\partial x}\left(Q\frac{E_n{}^m}{E_n}E_n\right) - \frac{\partial}{\partial x}\left(K_x A\frac{\partial (E_n{}^m/E_n)}{\partial x}E_n\right)\right] dx$$
$$+\int_{x_1}^{x_N} N_i \left(L_{HSn}\frac{E_n{}^m}{E_n} + \frac{\partial A}{\partial t}\right) E_n dx = \int_{x_1}^{x_N} N_i (R_{HSn} + AR_{E_n}) dx \tag{3.5.7.1.13}$$

Integrating by parts, we obtain

$$\int_{x_1}^{x_N} N_i A\frac{\partial E_n}{\partial t}dx + \int_{x_1}^{x_N} \frac{dN_i}{dx}\left(K_x A\frac{E_n{}^m}{E_n}\right)\frac{\partial E_n}{\partial x}dx - \int_{x_1}^{x_N}\frac{dW_i}{dx}\left(Q\frac{E_n{}^m}{E_n}\right) E_n dx + \int_{x_1}^{x_N}\frac{dW_i}{dx}\left(K_x A\frac{\partial (E_n{}^m/E_n)}{\partial x}\right) E_n dx$$
$$+\int_{x_1}^{x_N} N_i\left(L_{HSn}\frac{E_n{}^m}{E_n}+\frac{\partial A}{\partial t}\right) E_n dx = \int_{x_1}^{x_N} N_i\left(R_{HSn}+AR_{E_n}\right) dx + N_i K_x A\frac{E_n{}^m}{E_n}\frac{\partial E_n}{\partial x}\bigg|_{B1}^{B2} - W_i Q E_n{}^m\Big|_{B1}^{B2} + W_i K_x A\frac{\partial (E_n{}^m/E_n)}{\partial x}E_n\bigg|_{B1}^{B2} \tag{3.5.7.1.14}$$

Approximate solution E_n by a linear combination of the base functions as follows

$$E_n \approx \hat{E}_n = \sum_{j=1}^{N} E_{nj}(t) N_j(x) \tag{3.5.7.1.15}$$

Substituting Equation (3.5.7.1.15) into Equation (3.5.7.1.14), we obtain

$$\sum_{j=1}^{N}\left\{\begin{bmatrix} -\int_{x_1}^{x_N}\frac{dW_i}{dx}\left(Q\frac{E_n{}^m}{E_n}\right) N_j dx + \int_{x_1}^{x_N}\frac{dW_i}{dx}\left(K_x A\frac{\partial (E_n{}^m/E_n)}{\partial x}\right) N_j dx \\ +\int_{x_1}^{x_N}\frac{dN_i}{dx}\left(K_x A\frac{E_n{}^m}{E_n}\right)\frac{dN_j}{dx}dx + \int_{x_1}^{x_N} N_i\left(L_{HSn}\frac{E_n{}^m}{E_n}+\frac{\partial A}{\partial t}\right) N_j dx \end{bmatrix} E_{nj}(t)\right\} + \sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N} N_i A N_j dx\right)\frac{\partial E_{nj}(t)}{\partial t}\right] =$$
$$\int_{x_1}^{x_N} N_i\left(R_{HS}+AR_{E_n}\right) dx - \sum n\left[-N_i\left(K_x A\frac{E_n{}^m}{E_n}\right)\frac{\partial E_n}{\partial x} + W_i Q E_n{}^m - W_i\left(K_x A\frac{\partial (E_n{}^m/E_n)}{\partial x}\right) E_n\right]_b \tag{3.5.7.1.16}$$

Equation (3.5.7.1.16) can be written in matrix form as

$$\left([L1]+[L2]+[L3]+[L4]\right)\{E_n\} + [M]\left\{\frac{\partial E_n}{\partial t}\right\} = \{S\}+\{B\} \tag{3.5.7.1.17}$$

The matrices $[L1]$, $[L2]$, $[L3]$, $[L4]$, $[M]$ and load vectors $\{S\}$, $\{B\}$ are given by

$$L1_{ij} = -\int_{x_1}^{x_N}\frac{dW_i}{dx}\left(Q\frac{E_n{}^m}{E_n}\right) N_j dx \tag{3.5.7.1.18}$$

$$L2_{ij} = \int_{x_1}^{x_N}\frac{dW_i}{dx}\left(K_x A\frac{\partial (E_n{}^m/E_n)}{\partial x}\right) N_j dx \tag{3.5.7.1.19}$$

$$L3_{ij} = \int_{x_1}^{x_N}\frac{dN_i}{dx}\left(K_x A\frac{E_n{}^m}{E_n}\right)\frac{dN_j}{dx}dx \tag{3.5.7.1.20}$$

$$L4_{ij} = \int_{x_1}^{x_N} N_i\left(L_{HSn}\frac{E_n{}^m}{E_n}+\frac{\partial A}{\partial t}\right) N_j dx \tag{3.5.7.1.21}$$

$$M_{ij} = \int_{x_1}^{x_N} N_i A N_j dx \tag{3.5.7.1.22}$$

$$S_i = \int_{x_1}^{x_N} N_i \left(R_{HSn} + AR_{E_n} \right) dx \quad \textbf{(3.5.7.1.23)}$$

$$B_i = -n \left[W_i Q E_n^{\ m} - W_i \left(K_x A \frac{\partial (E_n^{\ m} / E_n)}{\partial x} \right) E_n - N_i \left(K_x A \frac{E_n^{\ m}}{E_n} \right) \frac{\partial E_n}{\partial x} \right]_b \quad \textbf{(3.5.7.1.24)}$$

To calculate [$L2$] through equation (3.5.7.1.19), assign

$$PPX = \frac{\partial (E_n^{\ m} / E_n)}{\partial x} \quad \textbf{(3.5.7.1.25)}$$

Then

$$\int_{x_1}^{x_N} N_i PPX dx = \int_{x_1}^{x_N} N_i \frac{\partial (E_n^{\ m} / E_n)}{\partial x} dx \quad \textbf{(3.5.7.1.26)}$$

$$\sum_{j=1}^{N} \left[\left(\int_{x_1}^{x_N} N_i N_j dx \right) PPX_j \right] = \sum_{j=1}^{N} \left[\left(\int_{x_1}^{x_N} N_i \frac{dN_j}{dx} dx \right) \left(\frac{E_n^{\ m}}{E_n} \right)_j \right] \quad \textbf{(3.5.7.1.27)}$$

So that

$$[QP1]\left\{ PPX \right\} = [QP2] \left\{ \frac{E_n^{\ m}}{E_n} \right\} \quad \textbf{(3.5.7.1.28)}$$

Lump [$QP1$] into diagonal matrix and assign

$$QP_{ij} = QP2_{ij} / QP1_{ii} \quad \textbf{(3.5.7.1.29)}$$

Then

$$\left\{ PPX \right\} = [QP] \left\{ \frac{E_n^{\ m}}{E_n} \right\} \quad \textbf{(3.5.7.1.30)}$$

Equation (3.5.7.1.17) can be simplified as

$$[L]\left\{ E_n \right\} + [M] \left\{ \frac{\partial E_n}{\partial t} \right\} = \left\{ S \right\} + \left\{ B \right\}, \ \ where [L] = [L1] + [L2] + [L3] + [L4] \quad \textbf{(3.5.7.1.31)}$$

Further,

$$[L]\left\{ W_1 E_n^{\ n+1/2} + W_2 E_n^{\ n} \right\} + \frac{[M]}{\Delta t} \left\{ E_n^{\ n+1/2} - E_n^{\ n} \right\} = \left\{ S \right\} + \left\{ B \right\} \quad \textbf{(3.5.7.1.32)}$$

So that

$$[CMATRX]\left\{ E_n^{\ n+1/2} \right\} = \{RLD\} \quad \textbf{(3.5.7.1.33)}$$

where

$$[CMATRX] = \frac{[M]}{\Delta t} + W_1 * [L] \quad \textbf{(3.5.7.1.34)}$$

$$\{RLD\} = \left(\frac{[M]}{\Delta t} - W_2[L]\right)\left\{E_n^{\ n}\right\} + \{S\} + \{B\} \tag{3.5.7.1.35}$$

The above equations are used to solve for the kinetic variable concentration at interior nodes, where the boundary term $\{B\}$ is zero.

The equation employed to determine the kinetic variable at junctions can be derived based on the conservation law of material mass and written as follows.

$$\mathcal{V}_j \frac{d(E_n)_j}{dt} + \frac{d\mathcal{V}_j}{dt}(E_n)_j = (M_{E_n}{}^{as})_j + (M_{E_n}{}^{rs})_j + (M_{E_n}{}^{os})_j + (M_{E_n}{}^{is})_j + \mathcal{V}_j (R_{E_n})_j + \sum_{k=1}^{NJRTH_j} Flux_k \tag{3.5.7.1.36}$$

where $\mathcal{V}_j$ is the junction volume, $(E_n)_j$ is the concentration of the n-th kinetic variable at Junction j, $(M_{E_n}{}^{as})_j$ is the rate of artificial source of E_n at Junction j, $(M_{E_n}{}^{rs})_j$ is the rate of rainfall source at Junction j, $(M_{E_n}{}^{os})_j$ is the rate of overland source at Junction j, $(M_{E_n}{}^{is})_j$ is exfiltration source at the junction, $(R_{E_n})_j$ is the rate kinetic variable concentration change due to reactions at the junction, $NJTRH_j$ is the number of river/stream reaches connected to the junction, and $Flux_k$ is the material flux of the kinetic variable contributed from the k-th reach to the junction.

$$Flux_k = n\left[Q^k (E_n^{\ m})^k - K_x A \frac{\partial (E_n^{\ m})^k}{\partial x}\right] \tag{3.5.7.1.37}$$

At n+1-th time step, equation (3.5.7.1.36) is approximated by

$$\mathcal{V}_j \frac{(E_n)_j^{\ n+1} - (E_n)_j^{\ n}}{\Delta t} + \frac{d\mathcal{V}_j}{dt}(E_n)_j = (M_{E_n}{}^{as})_j + (M_{E_n}{}^{rs})_j + (M_{E_n}{}^{os})_j + (M_{E_n}{}^{is})_j + \mathcal{V}_j (R_{E_n})_j + \sum_{k=1}^{NJRTH_j} Flux_k \tag{3.5.7.1.38}$$

which can be separated into two equations, according to Fully-implicit scheme, as follows

$$\mathcal{V}_j \frac{(E_n)_j^{\ n+1/2} - (E_n)_j^{\ n}}{\Delta t} + \frac{d\mathcal{V}_j}{dt}(E_n)_j = (M_{E_n}{}^{as})_j + (M_{E_n}{}^{rs})_j + (M_{E_n}{}^{os})_j + (M_{E_n}{}^{is})_j + \mathcal{V}_j (R_{E_n})_j + \sum_{k=1}^{NJRTH_j} Flux_k \tag{3.5.7.1.39}$$

$$\frac{(E_n)_j^{\ n+1} - (E_n)_j^{\ n+1/2}}{\Delta t} = 0 \tag{3.5.7.1.40}$$

First, solve equation (3.5.7.1.39) and get $(E_n)_j^{n+1/2}$. Second, solve equation (3.5.7.1.40) together with algebraic equations for equilibrium reactions using BIOGEOCHEM scheme to get the individual species concentration.

To solve equation (3.5.7.1.39), assign

$$(L_{HSn})_j = \frac{\mathcal{V}_j}{\Delta t} + \frac{\partial \mathcal{V}}{\partial t} \tag{3.5.7.1.41}$$

$$(R_{HSn})_j = \frac{\mathcal{V}_j^{\ n}(E_n)_j^{\ n}}{\Delta t} + W_2 (R_{HSn})_j^{\ n} + \mathcal{V}_j (R_{E_n})_j \tag{3.5.7.1.42}$$

$$Flux_k = W_1 \cdot Flux_k^{\ n+1} + W_2 \cdot Flux_k^{\ n} \tag{3.5.7.1.43}$$

Continue the calculation as follows

$$(M_{E_n}{}^{as})_j = \begin{cases} (S_S)_j * (E_{n^{as}})_j, \ \ if \ (S_S)_j > 0 \ \Rightarrow \ (R_{HSn})_j = (R_{HSn})_j + W_1 (S_S)_j * (E_{n^{as}})_j \\ (S_S)_j * (E_n{}^m)_j, \ \ if \ (S_S)_j \le 0 \ \Rightarrow \ (L_{HS_n})_j = (L_{HSn})_j - W_1 (S_S)_j * E_n{}^m / E_n \end{cases} \qquad \textbf{(3.5.7.1.44)}$$

$$(M_{E_n}{}^{os})_j = \begin{cases} (S_{os})_j * (E_{n^{os}})_j, \ \ if \ (S_{os})_j > 0 \ \Rightarrow \ (R_{HSn})_j = (R_{HSn})_j + W_1 (S_{os})_j * (E_{n^{os}})_j \\ (S_{os})_j * (E_n{}^m)_j, \ \ if \ (S_{os})_j \le 0 \ \Rightarrow \ (L_{HSn})_j = (L_{HSn})_j - W_1 (S_{os})_j * E_n{}^m / E_n \end{cases} \qquad \textbf{(3.5.7.1.45)}$$

where $(S_{os})_j$ is the flow rate of overland source to Junction *j* and $(E_{n^{os}})_j$ is the concentration of E_n in the overland source into Junction *j*.

$$(M_{E_n}{}^{rs})_j = \begin{cases} (S_R)_j * (E_{n^{rs}})_j, \ \ if \ (S_R)_j > 0 \ \Rightarrow \ (R_{HSn})_j = (R_{HSn})_j + W_1 (S_R)_j * (R_n)_j \\ (S_R)_j * (E_n{}^m)_j, \ \ if \ (S_R)_j \le 0 \ \Rightarrow \ (L_{HSn})_j = (L_{HSn})_j - W_1 (S_R)_j * E_n{}^m / E_n \end{cases} \qquad \textbf{(3.5.7.1.46)}$$

$$(M_{E_n}{}^{is})_j = \begin{cases} (S_I)_j * (E_{n^{is}})_j, \ \ if \ (S_I)_j > 0 \ \Rightarrow \ (R_{HSn})_j = (R_{HSn})_j + W_1 (S_I)_j * (E_{n^{is}})_j \\ (S_I)_j * (E_n{}^m)_j, \ \ if \ (S_I)_j \le 0 \ \Rightarrow \ (L_{HSn})_j = (L_{HSn})_j - W_1 (S_I)_j * E_n{}^m / E_n \end{cases} \qquad \textbf{(3.5.7.1.47)}$$

Then equation (3.5.7.1.39) is approximated by

$$(L_{HSn})_j (E_n)_j - \sum_{k=1}^{NJRTH_j} Flux_k = (R_{HSn})_j \qquad \textbf{(3.5.7.1.48)}$$

Assign

$$\{RLDW\} = \left(\frac{[M]}{\Delta t} - W_2 * [L] \right) \left\{ E_n{}^n \right\} + \left\{ S \right\} \qquad \textbf{(3.5.7.1.49)}$$

Equation (3.5.7.1.33) is modified as

$$[CMATRX] \left\{ E_n{}^{n+1/2} \right\} + \{Flux\} = \{RLDW\} \qquad \textbf{(3.5.7.1.50)}$$

The flux term in both equations (3.5.7.1.48) and (3.5.7.1.50) is specified as follows.

If $nQ > 0$, flow is going from reach to the junction

$$Flux_k = Q^k (E_n{}^m)^k = W_1 (Q^k)^{n+1} \frac{[(E_n{}^m)^k]^{n+1/2}}{[(E_n)^k]^{n+1/2}} [(E_n)^k]^{n+1/2} + W_2 (Q^k)^n [(E_n{}^m)^k]^n \qquad \textbf{(3.5.7.1.51)}$$

where the superscript n denotes the old time step, the superscript $n+1/2$ denotes the intermediate time step, $Flux_k$ is the flux of the *n*-th kinetic variable from the *k*-th reach to Junction *j*, Q^k is the flow rate from the *k*-th reach to Junction *j*, $(E_n)^k$ is the concentration of the *n*-th kinetic variable of the *k*-th reach, and $(E_n^m)^k$ is the mobile concentration of the *n*-th kinetic variable of the *k*-th reach.

If $nQ < 0$, flow is going from junction to the reach,

$$Flux_k = -Q^k (E_n{}^m)_j = -W_1 (Q^k)^{n+1} \frac{[(E_n{}^m)_j]^{n+1/2}}{[(E_n)_j]^{n+1/2}} [(E_n)_j]^{n+1/2} - W_2 (Q^k)^n [(E_n{}^m)_j]^n \quad \textbf{(3.5.7.1.52)}$$

So that equations (3.5.7.1.48) and (3.5.7.1.50) become a set of equation of $(E_n)_j$ and $(E_n)^k$.

For boundary node $i = b$ (use B as the input boundary value), the boundary term $\{B\}$ should be continuously calculated as follows.

$$\begin{aligned} B_i &= -n\left[W_i Q E_n{}^m - N_i K_x A \frac{E_n{}^m}{E_n} \frac{\partial E_n}{\partial x} - W_i K_x A \frac{\partial (E_n{}^m / E_n)}{\partial x} E_n \right]_b \\ &= -n\left[Q E_n{}^m - K_x A \frac{E_n{}^m}{E_n} \frac{\partial E_n}{\partial x} - K_x A \frac{\partial (E_n{}^m / E_n)}{\partial x} E_n \right]_b = -n\left(Q E_n{}^m - K_x A \frac{\partial E_n{}^m}{\partial x} \right)_b \end{aligned} \quad \textbf{(3.5.7.1.53)}$$

Dirichlet boundary condition

$$E_n{}^m = E_n{}^m (x_b, t) \quad \textbf{(3.5.7.1.54)}$$

Variable boundary condition

When flow is coming in from outside ($nQ < 0$)

$$n\left(Q E_n{}^m - A K_x \frac{\partial E_n{}^m}{\partial x} \right) = n Q E_n{}^m (x_b, t) \;\; \Rightarrow \;\; B_i = -n Q E_n{}^m (x_b, t) \quad \textbf{(3.5.7.1.55)}$$

When Flow is going out from inside ($nQ > 0$)

$$-n A K_x \frac{\partial E_n{}^m}{\partial x} = 0 \;\; \Rightarrow \;\; B_i = -n Q E_n{}^m \quad \textbf{(3.5.7.1.56)}$$

Cauchy boundary condition

$$n\left(Q E_n{}^m - A K_x \frac{\partial E_n{}^m}{\partial x} \right) = Q_{En} (x_b, t) \;\; \Rightarrow \;\; B_i = -Q_{En} (x_b, t) \quad \textbf{(3.5.7.1.57)}$$

Neumann boundary condition

$$-n A K_x \frac{\partial E_n{}^m}{\partial x} = Q_{En} (x_b, t) \;\; \Rightarrow \;\; B_i = -n Q E_n{}^m - Q_{En} (x_b, t) \quad \textbf{(3.5.7.1.58)}$$

3.5.7.2 Mixed Predictor-corrector/Operator-Splitting Scheme

Recall the continuity equation for kinetic-variables, equation (3.5.7.1.1), as follows.

$$A \frac{\partial E_n}{\partial t} + \frac{\partial A}{\partial t} E_n + \frac{\partial (Q E_n{}^m)}{\partial x} - \frac{\partial}{\partial x}\left(K_x A \frac{\partial E_n{}^m}{\partial x} \right) = M_{E_n}{}^{as} + M_{E_n}{}^{rs} + M_{E_n}{}^{is} + M_{E_n}{}^{os1} + M_{E_n}{}^{os2} + AR_{E_n} \quad \textbf{(3.5.7.2.1)}$$

At (n+1)-th time step, equation (3.5.7.2.1) is approximated by

$$A\frac{(E_n)^{n+1}-(E_n)^n}{\Delta t}+\frac{\partial A}{\partial t}E_n+\frac{\partial(QE_n{}^m)}{\partial x}-\frac{\partial}{\partial x}\left(K_xA\frac{\partial E_n{}^m}{\partial x}\right)=M_{E_n}{}^{as}+M_{E_n}{}^{rs}+M_{E_n}{}^{is}+M_{E_n}{}^{os1}+M_{E_n}{}^{os2}+AR_{E_n} \quad \textbf{(3.5.7.2.2)}$$

According to Mixed Predictor-corrector/Operator-Splitting Scheme, equation (3.5.7.2.2) can be separated into two equations as follows

$$\begin{aligned}&A\frac{(E_n{}^m)^{n+1/2}-(E_n{}^m)^n}{\Delta t}+\frac{\partial A}{\partial t}E_n^m+\frac{\partial(QE_n{}^m)}{\partial x}-\frac{\partial}{\partial x}\left(K_xA\frac{\partial E_n{}^m}{\partial x}\right)=M_{E_n}{}^{as}+M_{E_n}{}^{rs}+M_{E_n}{}^{os1}+M_{E_n}{}^{os2}+M_{E_n}{}^{is}\\&+AR_{E_n}{}^n-A\frac{\partial(\ell nA)}{\partial t}(E_n^{im})^n\end{aligned} \quad \textbf{(3.5.7.2.3)}$$

$$\frac{E_n{}^{n+1}-[(E_n{}^m)^{n+1/2}+(E_n{}^{im})^n]}{\Delta t}=R_{E_n}{}^{n+1}-R_{E_n}{}^n-\frac{\partial(\ell nA)}{\partial t}(E_n^{im})^{n+1}+\frac{\partial(\ell nA)}{\partial t}(E_n^{im})^n \quad \textbf{(3.5.7.2.4)}$$

First, solve equation (3.5.7.2.3) and obtain $(E_n{}^m)^{n+1/2}$. Second, solve equation (3.5.7.2.4) together with algebraic equations for equilibrium reactions using BIOGEOCHEM scheme to obtain $(E_n)^{n+1}$ and the individual species concentration.

To solve equation (3.5.7.2.3), assign and calculate R_{HSn} and L_{HSn} same as that in section (3.5.7.1). Then equation (3.5.7.2.3) is simplified as

$$A\frac{(E_n{}^m)^{n+1/2}-(E_n{}^m)^n}{\Delta t}+\frac{\partial A}{\partial t}E_n^m+\frac{\partial(QE_n{}^m)}{\partial x}-\frac{\partial}{\partial x}\left(K_xA\frac{\partial E_n{}^m}{\partial x}\right)+L_{HSn}E_n{}^m=R_{HSn}+AR_{E_n}{}^n-\frac{\partial A}{\partial t}(E_n^{im})^n \quad \textbf{(3.5.7.2.5)}$$

Use Galerkin or Petrov-Galerkin FEM for the spatial discretization of transport equations. Integrate Equation (3.5.7.2.5) in the spatial dimensions over the entire region as follows.

$$\begin{aligned}&\int_{x_1}^{x_N}N_i\left[A\frac{\partial E_n{}^m}{\partial t}-\frac{\partial}{\partial x}\left(K_xA\frac{\partial E_n{}^m}{\partial x}\right)\right]dx+\int_{x_1}^{x_N}W_i\frac{\partial\left(QE_n{}^m\right)}{\partial x}dx+\int_{x_1}^{x_N}N_i\left(L_{HSn}+\frac{\partial A}{\partial t}\right)E_n{}^m dx=\\&\int_{x_1}^{x_N}N_i\left(R_{HSn}+AR_{E_n}{}^n-\frac{\partial A}{\partial t}(E_n^{im})^n\right)dx\end{aligned} \quad \textbf{(3.5.7.2.6)}$$

Integrating by parts, we obtain

$$\begin{aligned}&\int_{x_1}^{x_N}N_iA\frac{\partial E_n{}^m}{\partial t}dx+\int_{x_1}^{x_N}\frac{dN_i}{dx}K_xA\frac{\partial E_n{}^m}{\partial x}dx-\int_{x_1}^{x_N}\frac{dW_i}{dx}QE_n{}^m dx+\int_{x_1}^{x_N}N_i\left(L_{HSn}+\frac{\partial A}{\partial t}\right)E_n{}^m dx\\&=\int_{x_1}^{x_N}N_i\left(R_{HSn}+AR_{E_n}{}^n-\frac{\partial A}{\partial t}(E_n^{im})^n\right)dx-W_iQE_n{}^m\Big|_{B1}^{B2}+N_iK_xA\frac{\partial E_n{}^m}{\partial x}\Big|_{B1}^{B2}\end{aligned} \quad \textbf{(3.5.7.2.7)}$$

Approximate solution $E_n{}^m$ by a linear combination of the base functions as follows

$$E_n{}^m\approx\hat{E}_n{}^m=\sum_{j=1}^{N}E_{nj}{}^m(t)N_j(x) \quad \textbf{(3.5.7.2.8)}$$

Substituting Equation (3.5.7.2.8) into Equation (3.5.7.2.7), we obtain

$$\sum_{j=1}^{N}\left[\left(-\int_{x_1}^{x_N}\frac{dW_i}{dx}QN_j dx+\int_{x_1}^{x_N}\frac{dN_i}{dx}K_x A\frac{dN_j}{dx}dx+\int_{x_1}^{x_N}N_i\left(L_{HSn}+\frac{\partial A}{\partial t}\right)N_j dx\right)E_{nj}{}^m(t)\right]$$
$$+\sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N}N_i AN_j dx\right)\frac{\partial E_{nj}{}^m(t)}{\partial t}\right]=\int_{x_1}^{x_N}N_i\left(R_{HSn}+AR_{E_n}{}^n-\frac{\partial A}{\partial t}(E_n^{im})^n\right)dx-\sum n\left[W_iQE_n{}^m-N_iK_xA\frac{\partial E_n{}^m}{\partial x}\right]_b \quad \textbf{(3.5.7.2.9)}$$

Equation (3.5.7.2.9) can be written in matrix form as

$$([L1]+[L2]+[L3])\left\{E_n{}^m\right\}+[M]\left\{\frac{\partial E_n{}^m}{\partial t}\right\}=\{S\}+\{B\} \quad \textbf{(3.5.7.2.10)}$$

The matrices $[L1]$, $[L2]$, $[L3]$, $[M]$ and load vectors $\{S\}$, $\{B\}$ are given by

$$L1_{ij}=-\int_{x_1}^{x_N}\frac{dW_i}{dx}QN_j dx \quad \textbf{(3.5.7.2.11)}$$

$$L2_{ij}=\int_{x_1}^{x_N}\frac{dN_i}{dx}K_xA\frac{dN_j}{dx}dx \quad \textbf{(3.5.7.2.12)}$$

$$L3_{ij}=\int_{x_1}^{x_N}N_i\left(L_{HSn}+\frac{\partial A}{\partial t}\right)N_j dx \quad \textbf{(3.5.7.2.13)}$$

$$M_{ij}=\int_{x_1}^{x_N}N_iAN_j dx \quad \textbf{(3.5.7.2.14)}$$

$$S_i=\int_{x_1}^{x_N}N_i\left(R_{HSn}+AR_{E_n}{}^n-\frac{\partial A}{\partial t}(E_n^{im})^n\right)dx \quad \textbf{(3.5.7.2.15)}$$

$$B_i=-n\left(W_iQE_n{}^m-N_iK_xA\frac{\partial E_n{}^m}{\partial x}\right)_b \quad \textbf{(3.5.7.2.16)}$$

where all the terms listed above are calculated with the corresponding time weighting values. Equation (3.5.7.2.10) is then simplified as

$$[L]\left\{E_n{}^m\right\}+[M]\left\{\frac{\partial E_n{}^m}{\partial t}\right\}=\{S\}+\{B\},\ where[L]=[L1]+[L2]+[L3] \quad \textbf{(3.5.7.2.17)}$$

Further,

$$[L]\left\{W_1*(E_n{}^m)^{n+1/2}+W_2*(E_n{}^m)^n\right\}+[M]\left\{\frac{(E_n{}^m)^{n+1/2}-(E_n{}^m)^n}{\Delta t}\right\}=\{S\}+\{B\} \quad \textbf{(3.5.7.2.18)}$$

So that

$$[CMATRX]\left\{(E_n{}^m)^{n+1/2}\right\}=\{RLD\} \quad \textbf{(3.5.7.2.19)}$$

where

$$[CMATRX]=\frac{[M]}{\Delta t}+W_1*[L] \quad \textbf{(3.5.7.2.20)}$$

$$\{RLD\} = \left(\frac{[M]}{\Delta t} - W_2 * [L] \right) \left\{ (E_n{}^m)^n \right\} + \{S\} + \{B\} \tag{3.5.7.2.21}$$

The above equations are used to solve for the kinetic variable concentration at interior nodes, where the boundary term {B} is zero.

For junction nodes, recall equation (3.5.7.1.38) as follows.

$$\mathcal{V}_j \frac{(E_n)_j{}^{n+1} - (E_n)_j{}^n}{\Delta t} + \frac{d\mathcal{V}_j}{dt}(E_n)_j = (M_{E_n}{}^{as})_j + (M_{E_n}{}^{rs})_j + (M_{E_n}{}^{os})_j + (M_{E_n}{}^{is})_j + \mathcal{V}_j (R_{E_n})_j + \sum_{k=1}^{NJRTH_j} Flux_k \tag{3.5.7.2.22}$$

which can be separated into two equations, according to mixed Predictor-corrector/operator-splitting scheme, as follows

$$\begin{aligned} &\mathcal{V}_j \frac{(E_n^m)_j{}^{n+1/2} - (E_n^m)_j{}^n}{\Delta t} + \frac{d\mathcal{V}_j}{dt}(E_n^m)_j = (M_{E_n}{}^{as})_j + (M_{E_n}{}^{rs})_j + (M_{E_n}{}^{os})_j + (M_{E_n}{}^{is})_j + \\ &\mathcal{V}_j (R_{E_n})_j{}^n - \frac{d\mathcal{V}_j}{dt}(E_n^{im})_j{}^n + \sum_{k=1}^{NJRTH_j} Flux_k \end{aligned} \tag{3.5.7.2.23}$$

$$\frac{(E_n)_j{}^{n+1} - [(E_n{}^m)_j{}^{n+1/2} + (E_n{}^{im})_j{}^n]}{\Delta t} = \mathcal{V}_j (R_{E_n})_j{}^{n+1} - \mathcal{V}_j (R_{E_n})_j{}^n - \frac{\partial(\ell n \mathcal{V}_j)}{\partial t}(E_n^{im})_j{}^{n+1} + \frac{\partial(\ell n \mathcal{V}_j)}{\partial t}(E_n^{im})_j{}^n \tag{3.5.7.2.24}$$

First, solve equation (3.5.7.2.23) and get $(E_n{}^m)_j{}^{n+1/2}$. Second, solve equation (3.5.7.2.24) together with algebraic equations for equilibrium reactions using BIOGEOCHEM scheme to obtain the individual species concentration.

To solve equation (3.5.7.2.23), assign

$$(L_{HSn})_j = \frac{\mathcal{V}_j^n}{\Delta t} + \frac{d\mathcal{V}_j}{dt} \tag{3.5.7.2.25}$$

$$(R_{HSn})_j = \frac{\mathcal{V}_j^n (E_n{}^m)_j{}^n}{\Delta t} + W_2 (R_{HSn})_j{}^n + \mathcal{V}_j (R_{E_n})_j{}^n - \frac{d\mathcal{V}_j}{dt}(E_n^{im})_j{}^n \tag{3.5.7.2.26}$$

$$Flux_k = W_1 \cdot Flux_k{}^{n+1} + W_2 \cdot Flux_k{}^n \tag{3.5.7.2.27}$$

Continue the calculation as follows

$$(M_{E_n}{}^{as})_j = \begin{cases} (S_S)_j * (E_{n^{as}})_j, \ \ if \ (S_S)_j > 0 \ \Rightarrow \ (R_{HSn})_j = (R_{HSn})_j + W_1 (S_S)_j * (E_{n^{as}})_j \\ (S_S)_j * (E_n{}^m)_j, \ \ if \ (S_S)_j \le 0 \ \Rightarrow \ (L_{HSn})_j = (L_{HSn})_j - W_1 (S_S)_j \end{cases} \tag{3.5.7.2.28}$$

$$(M_{E_n}{}^{os})_j = \begin{cases} (S_{os})_j * (E_{n^{os}})_j, \ \ if \ (S_{os})_j > 0 \ \Rightarrow \ (R_{HSn})_j = (R_{HSn})_j + W_1 (S_{os})_j * (E_{n^{os}})_j \\ (S_{os})_j * (E_n{}^m)_j, \ \ if \ (S_{os})_j \le 0 \ \Rightarrow \ (L_{HSn})_j = (L_{HSn})_j - W_1 (S)_j \end{cases} \tag{3.5.7.2.29}$$

$$(M_{E_n}{}^{rs})_j = \begin{cases} (S_R)_j * (E_{n^{rs}})_j, \ \ if \ (S_R)_j > 0 \ \Rightarrow \ (R_{HSn})_j = (R_{HSn})_j + W_1 (S_R)_j * (E_{n^{rs}})_j \\ (S_R)_j * (E_n{}^m)_j, \ \ if \ (S_R)_j \le 0 \ \Rightarrow \ (L_{HSn})_j = (L_{HSn})_j - W_1 (S_R)_j \end{cases} \tag{3.5.7.2.30}$$

$$(M_{E_n}{}^{is})_j = \begin{cases} (S_I)_j *(E_{n^{is}})_j, \;\; if \;\; (S_I)_j > 0 \;\; \Rightarrow \;\; (R_{HSn})_j = (R_{HSn})_j + W_1 (S_I)_j *(E_{n^{is}})_j \\ (S_I)_j *(E_n{}^m)_j, \;\; if \;\; (S_I)_j \le 0 \;\; \Rightarrow \;\; (L_{HSn})_j = (L_{HSn})_j - W_1 (S_I)_j \end{cases} \quad \textbf{(3.5.7.2.31)}$$

Then equation (3.5.7.2.23) is approximated by

$$(L_{HSn})_j (E_n{}^m)_j - \sum_{k=1}^{NJRTH_j} Flux_k = (R_{HSn})_j \quad \textbf{(3.5.7.2.32)}$$

Assign

$$\{RLDW\} = \left(\frac{[M]}{\Delta t} - W_2 * [L] \right) \left\{ (E_n{}^m)^n \right\} + \{S\} \quad \textbf{(3.5.7.2.33)}$$

Equation (3.5.7.2.19) is modified as

$$[CMATRX] \left\{ (E_n{}^m)^{n+1/2} \right\} + \{Flux\} = \{RLDW\} \quad \textbf{(3.5.7.2.34)}$$

The flux term in both equations (3.5.7.2.32) and (3.5.7.2.34) is specified as follows.

If $nQ > 0$, flow is going from reach to the junction

$$Flux_k = Q^k (E_n{}^m)^k = W_1 (Q^k)^{n+1} [(E_n{}^m)^k]^{n+1/2} + W_2 (Q^k)^n [(E_n{}^m)^k]^n \quad \textbf{(3.5.7.2.35)}$$

If $nQ < 0$, flow is going from junction to the reach,

$$Flux_k = -Q^k (E_n{}^m)_j = -W_1 (Q^k)^{n+1} [(E_n{}^m)_j]^{n+1/2} - W_2 (Q^k)^n [(E_n{}^m)_j]^n \quad \textbf{(3.5.7.2.36)}$$

So that equations (3.5.7.2.32) and (3.5.7.2.34) become a set of equations of $(E_n{}^m)_j$ and $(E_n{}^m)^k$.

For boundary node $i = b$, the boundary term $\{B\}$ should be continuously calculated same as that using Fully-implicit scheme in section 3.5.5.1.

3.5.7.3 Operator-splitting

Recall the continuity equation for kinetic-variables, equation (3.5.7.1.1), as follows.

$$A \frac{\partial E_n}{\partial t} + \frac{\partial A}{\partial t} E_n + \frac{\partial (Q E_n{}^m)}{\partial x} - \frac{\partial}{\partial x} \left(K_x A \frac{\partial E_n{}^m}{\partial x} \right) = M_{E_n}{}^{as} + M_{E_n}{}^{rs} + M_{E_n}{}^{is} + M_{E_n}{}^{os1} + M_{E_n}{}^{os2} + AR_{E_n} \quad \textbf{(3.5.7.3.1)}$$

At (n+1)-th time step, equation (3.5.7.3.1) is approximated by

$$A \frac{(E_n)^{n+1} - (E_n)^n}{\Delta t} + \frac{\partial A}{\partial t} E_n + \frac{\partial (Q E_n{}^m)}{\partial x} - \frac{\partial}{\partial x} \left(K_x A \frac{\partial E_n{}^m}{\partial x} \right) = M_{E_n}{}^{as} + M_{E_n}{}^{rs} + M_{E_n}{}^{is} + M_{E_n}{}^{os1} + M_{E_n}{}^{os2} + AR_{E_n} \quad \textbf{(3.5.7.3.2)}$$

According to Operator-splitting scheme, equation (3.5.7.3.2) can be separated into two equations as follows

$$A\frac{(E_n{}^m)^{n+1/2}-(E_n{}^m)^n}{\Delta t}+\frac{\partial A}{\partial t}E_n^m+\frac{\partial (QE_n{}^m)}{\partial x}-\frac{\partial}{\partial x}\left(K_x A\frac{\partial E_n{}^m}{\partial x}\right)=M_{E_n}{}^{as}+M_{E_n}{}^{rs}+M_{E_n}{}^{os1}+M_{E_n}{}^{os2}+M_{E_n}{}^{is} \quad \textbf{(3.5.7.3.3)}$$

$$\frac{E_n{}^{n+1}-[(E_n{}^m)^{n+1/2}+(E_n{}^{im})^n]}{\Delta t}=R_{E_n}{}^{n+1}-\frac{\partial(\ell nA)}{\partial t}(E_n^{im})^{n+1} \quad \textbf{(3.5.7.3.4)}$$

First, solve equation (3.5.7.3.3) and get $(E_n{}^m)^{n+1/2}$. Second, solve equation (3.5.7.3.4) together with algebraic equations for equilibrium reactions using BIOGEOCHEM scheme to obtain $(E_n)^{n+1}$ and the individual species concentration.

To solve equation (3.5.7.3.3), assign and calculate R_{HSn} and L_{HSn} same as that in section (3.5.7.1). Then equation (3.5.7.3.3) is simplified as

$$A\frac{(E_n{}^m)^{n+1/2}-(E_n{}^m)^n}{\Delta t}+\frac{\partial A}{\partial t}E_n^m+\frac{\partial (QE_n{}^m)}{\partial x}-\frac{\partial}{\partial x}\left(K_x A\frac{\partial E_n{}^m}{\partial x}\right)+\left(L_{HSn}+\frac{\partial A}{\partial t}\right)E_n{}^m=R_{HSn} \quad \textbf{(3.5.7.3.5)}$$

Use Galerkin or Petrov-Galerkin FEM for the spatial discretization of transport equations. Integrate Equation (3.5.7.3.5) in the spatial dimensions over the entire region as follows.

$$\int_{x_1}^{x_N} N_i\left[A\frac{\partial E_n{}^m}{\partial t}-\frac{\partial}{\partial x}\left(K_x A\frac{\partial E_n{}^m}{\partial x}\right)\right]dx+\int_{x_1}^{x_N} W_i\frac{\partial\left(QE_n{}^m\right)}{\partial x}dx+\int_{x_1}^{x_N} N_i\left(L_{HSn}+\frac{\partial A}{\partial t}\right)E_n{}^m dx=\int_{x_1}^{x_N} N_i R_{HSn}dx \quad \textbf{(3.5.7.3.6)}$$

Integrating by parts, we obtain

$$\begin{aligned}&\int_{x_1}^{x_N} N_i A\frac{\partial E_n{}^m}{\partial t}dx+\int_{x_1}^{x_N}\frac{dN_i}{dx}K_x A\frac{\partial E_n{}^m}{\partial x}dx-\int_{x_1}^{x_N}\frac{dW_i}{dx}QE_n{}^m dx+\int_{x_1}^{x_N} N_i\left(L_{HSn}+\frac{\partial A}{\partial t}\right)E_n{}^m dx\\&=\int_{x_1}^{x_N} N_i R_{HSn}dx-W_i QE_n{}^m\Big|_{B1}^{B2}+N_i K_x A\frac{\partial E_n{}^m}{\partial x}\Big|_{B1}^{B2}\end{aligned} \quad \textbf{(3.5.7.3.7)}$$

Approximate solution $E_n{}^m$ by a linear combination of the base functions as follows

$$E_n{}^m\approx\hat{E}_n{}^m=\sum_{j=1}^{N}E_{nj}{}^m(t)N_j(x) \quad \textbf{(3.5.7.3.8)}$$

Substituting Equation (3.5.7.3.8) into Equation (3.5.7.3.7), we obtain

$$\begin{aligned}&\sum_{j=1}^{N}\left[\left(-\int_{x_1}^{x_N}\frac{dW_i}{dx}QN_j dx+\int_{x_1}^{x_N}\frac{dN_i}{dx}K_x A\frac{dN_j}{dx}dx+\int_{x_1}^{x_N}N_i\left(L_{HSn}+\frac{\partial A}{\partial t}\right)N_j dx\right)E_{nj}{}^m(t)\right]\\&+\sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N}N_i AN_j dx\right)\frac{dE_{nj}{}^m(t)}{dt}\right]=\int_{x_1}^{x_N}N_i R_{HSn}dx-\sum n\left[W_i QE_n{}^m-N_i K_x A\frac{\partial E_n{}^m}{\partial x}\right]_b\end{aligned} \quad \textbf{(3.5.7.3.9)}$$

Equation (3.5.8.2.19) can be written in matrix form as

$$\left([L1]+[L2]+[L3]\right)\left\{E_n{}^m\right\}+[M]\left\{\frac{dE_n{}^m}{dt}\right\}=\left\{S\right\}+\left\{B\right\} \quad \textbf{(3.5.7.3.10)}$$

The matrices $[L1]$, $[L2]$, $[L3]$, $[M]$ and load vectors $\{S\}$, $\{B\}$ are given by

$$L1_{ij} = -\int_{x_1}^{x_N} \frac{dW_i}{dx} QN_j dx \qquad \textbf{(3.5.7.3.11)}$$

$$L2_{ij} = \int_{x_1}^{x_N} \frac{dN_i}{dx} K_x A \frac{dN_j}{dx} dx \qquad \textbf{(3.5.7.3.12)}$$

$$L3_{ij} = \int_{x_1}^{x_N} N_i \left(L_{HSn} + \frac{\partial A}{\partial t} \right) N_j dx \qquad \textbf{(3.5.7.3.13)}$$

$$M_{ij} = \int_{x_1}^{x_N} N_i A N_j dx \qquad \textbf{(3.5.7.3.14)}$$

$$S_i = \int_{x_1}^{x_N} N_i R_{HSn} dx \qquad \textbf{(3.5.7.3.15)}$$

$$B_i = -n \left(W_i Q E_n{}^m - N_i K_x A \frac{\partial E_n{}^m}{\partial x} \right)_b \qquad \textbf{(3.5.7.3.16)}$$

where all the terms listed above are calculated with the corresponding time weighting values.

Equation (3.5.7.2.10) is simplified as

$$[L]\left\{E_n{}^m\right\} + [M]\left\{\frac{dE_n{}^m}{dt}\right\} = \left\{S\right\} + \left\{B\right\}, \ \ where [L] = [L1] + [L2] + [L3] \qquad \textbf{(3.5.7.3.17)}$$

Further,

$$[L]\left\{W_1 * (E_n{}^m)^{n+1/2} + W_2 * (E_n{}^m)^n\right\} + [M]\left\{\frac{(E_n{}^m)^{n+1/2} - (E_n{}^m)^n}{\Delta t}\right\} = \left\{S\right\} + \left\{B\right\} \qquad \textbf{(3.5.7.3.18)}$$

So that

$$[CMATRX]\left\{(E_n{}^m)^{n+1/2}\right\} = \{RLD\} \qquad \textbf{(3.5.7.3.19)}$$

$$[CMATRX] = \frac{[M]}{\Delta t} + W_1 * [L] \qquad \textbf{(3.5.7.3.20)}$$

$$\{RLD\} = \left(\frac{[M]}{\Delta t} - W_2 * [L] \right) \left\{(E_n{}^m)^n\right\} + \left\{S\right\} + \left\{B\right\} \qquad \textbf{(3.5.7.3.21)}$$

The above equations are used to solve for the kinetic variable concentration at interior nodes, where the boundary term $\{B\}$ is zero.

For junction nodes, recall equation (3.5.7.2.22) as follows.

$$V_j \frac{(E_n)_j{}^{n+1} - (E_n)_j{}^n}{\Delta t} + \frac{dV_j}{dt}(E_n)_j = (M_{E_n}{}^{as})_j + (M_{E_n}{}^{rs})_j + (M_{E_n}{}^{os})_j + (M_{E_n}{}^{is})_j + V_j (R_{E_n})_j + \sum_{k=1}^{NJRTH_j} Flux_k \qquad \textbf{(3.5.7.3.22)}$$

which can be separated into two equations, according to Operator-splitting scheme, as follows

$$\mathcal{V}_j \frac{(E_n^m)_j^{n+1/2} - (E_n^m)_j^n}{\Delta t} + \frac{d\mathcal{V}_j}{dt}(E_n^m)_j = (M_{E_n}{}^{as})_j + (M_{E_n}{}^{rs})_j + (M_{E_n}{}^{os})_j + (M_{E_n}{}^{is})_j + \sum_{k=1}^{NJRTH_j} Flux_k \qquad \textbf{(3.5.7.3.23)}$$

$$\frac{(E_n)_j^{n+1} - [(E_n{}^m)_j^{n+1/2} + (E_n{}^{im})_j{}^n]}{\Delta t} = \mathcal{V}_j (R_{E_n})_j^{n+1} - \frac{\partial(\ell n \mathcal{V}_j)}{\partial t}(E_n^{im})_j^{n+1} \qquad \textbf{(3.5.7.3.24)}$$

First, solve equation (3.5.7.3.23) and get $(E_n{}^m)_j^{n+1/2}$. Second, solve equation (3.5.7.3.24) together with algebraic equations for equilibrium reactions using BIOGEOCHEM scheme to obtain the individual species concentration and $(E_n)_j^{n+1}$.

To solve equation (3.5.7.3.23), assign

$$(L_{HSn})_j = \frac{\mathcal{V}_j^n}{\Delta t} + \frac{d\mathcal{V}_j}{dt} \qquad \textbf{(3.5.7.3.25)}$$

$$(R_{HSn})_j = \frac{\mathcal{V}_j^n (E_n{}^m)_j{}^n}{\Delta t} + W_2 (R_{HSn})_j^n \qquad \textbf{(3.5.7.3.26)}$$

$$\mathrm{Flux_k = W_1 \cdot Flux_k{}^{n+1} + W_2 \cdot Flux_k{}^n} \qquad \textbf{(3.5.7.3.27)}$$

Continue the calculation as follows

$$(M_{E_n}{}^{as})_j = \begin{cases} (S_S)_j * (E_{n^{as}})_j, \ \ if \ \ (S_S)_j > 0 \ \Rightarrow \ (R_{HSn})_j = (R_{HSn})_j + W_1 (S_S)_j * (E_{n^{as}})_j \\ (S_S)_j * (E_n{}^m)_j, \ \ if \ \ (S_S)_j \le 0 \ \Rightarrow \ (L_{HSn})_j = (L_{HSn})_j - W_1 (S_S)_j \end{cases} \qquad \textbf{(3.5.7.3.28)}$$

$$(M_{E_n}{}^{os})_j = \begin{cases} (S_{os})_j * (E_{n^{os}})_j, \ \ if \ \ (S_{os})_j > 0 \ \Rightarrow \ (R_{HSn})_j = (R_{HSn})_j + W_1 (S_{os})_j * (E_{n^{os}})_j \\ (S_{os})_j * (E_n{}^m)_j, \ \ if \ \ (S_{os})_j \le 0 \ \Rightarrow \ (L_{HSn})_j = (L_{HSn})_j - W_1 (S)_j \end{cases} \qquad \textbf{(3.5.7.3.29)}$$

$$(M_{E_n}{}^{rs})_j = \begin{cases} (S_R)_j * (E_{n^{rs}})_j, \ \ if \ \ (S_R)_j > 0 \ \Rightarrow \ (R_{HSn})_j = (R_{HSn})_j + W_1 (S_R)_j * (E_{n^{rs}})_j \\ (S_R)_j * (E_n{}^m)_j, \ \ if \ \ (S_R)_j \le 0 \ \Rightarrow \ (L_{HSn})_j = (L_{HSn})_j - W_1 (S_R)_j \end{cases} \qquad \textbf{(3.5.7.3.30)}$$

$$(M_{E_n}{}^{is})_j = \begin{cases} (S_I)_j * (E_{n^{is}})_j, \ \ if \ \ (S_I)_j > 0 \ \Rightarrow \ (R_{HSn})_j = (R_{HSn})_j + W_1 (S_I)_j * (E_{n^{is}})_j \\ (S_I)_j * (E_n{}^m)_j, \ \ if \ \ (S_I)_j \le 0 \ \Rightarrow \ (L_{HSn})_j = (L_{HSn})_j - W_1 (S_I)_j \end{cases} \qquad \textbf{(3.5.7.3.31)}$$

Then equation (3.5.7.3.23) is approximated by

$$(L_{HSn})_j (E_n{}^m)_j - \sum_{k=1}^{NJRTH_j} Flux_k = (R_{HSn})_j \qquad \textbf{(3.5.7.3.32)}$$

Assign

$$\{RLDW\} = \left(\frac{[M]}{\Delta t} - W_2 * [L] \right) \left\{ (E_n{}^m)^n \right\} + \left\{ S \right\} \qquad \textbf{(3.5.7.3.33)}$$

Equation (3.5.7.3.19) is modified as

$$[CMATRX] \left\{ (E_n{}^m)^{n+1/2} \right\} + \{Flux\} = \{RLDW\} \qquad \textbf{(3.5.7.3.34)}$$

The flux term in both equation (3.5.7.3.32) and (3.5.7.3.34) is specified as follows.

If $nQ > 0$, flow is going from reach to the junction

$$Flux_k = Q^k (E_n{}^m)^k = W_1 (Q^k)^{n+1} [(E_n{}^m)^k]^{n+1/2} + W_2 (Q^k)^n [(E_n{}^m)^k]^n \tag{3.5.7.3.35}$$

If $nQ < 0$, flow is going from junction to the reach,

$$Flux_k = -Q^k (E_n{}^m)_j = -W_1 (Q^k)^{n+1} [(E_n{}^m)_j]^{n+1/2} - W_2 (Q^k)^n [(E_n{}^m)_j]^n \tag{3.5.7.3.36}$$

Equations (3.5.7.3.32) and (3.5.7.3.34) become a set of equation of $(E_n{}^m)_j$ and $(E_n{}^m)^k$.

For boundary node $i = b$, the boundary term {B} should be continuously calculated same as that using Fully-implicit scheme in section 3.5.5.1.

3.5.8 Finite Application of the Finite Element Method to the Advective Form of the Transport Equations to Solve 1-D Kinetic Variable

3.5.8.1 Fully-implicit scheme

Recall the continuity equation for kinetic-variables, equation (2.5.44), as follows.

$$A\frac{\partial E_n}{\partial t} + \frac{\partial A}{\partial t}E_n + \frac{\partial (QE_n{}^m)}{\partial x} - \frac{\partial}{\partial x}\left(K_x A \frac{\partial E_n{}^m}{\partial x}\right) = M_{E_n}{}^{as} + M_{E_n}{}^{rs} + M_{E_n}{}^{is} + M_{E_n}{}^{os1} + M_{E_n}{}^{os2} + AR_{E_n} \tag{3.5.8.1.1}$$

According to the governing equation of water flow in 1-D river/stream

$$\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} = S_S + S_R + S_I + S_1 + S_2 \tag{3.5.8.1.2}$$

Equation (3.5.8.1.1) can be modified as follows.

$$\begin{aligned} &A\frac{\partial E_n}{\partial t} + \frac{\partial A}{\partial t}E_n + Q\frac{\partial E_n{}^m}{\partial x} - \frac{\partial}{\partial x}\left(K_x A \frac{\partial E_n{}^m}{\partial x}\right) - \left[\frac{\partial A}{\partial t} - (S_S + S_R + S_I + S_1 + S_2)\right]E_n{}^m \\ &= M_{E_n}{}^{as} + M_{E_n}{}^{rs} + M_{E_n}{}^{is} + M_{E_n}{}^{os1} + M_{E_n}{}^{os2} + AR_{E_n} \end{aligned} \tag{3.5.8.1.3}$$

At n+1-th time step, equation (3.5.8.1.3) is approximated by

$$\begin{aligned} &A\frac{(E_n)^{n+1} - (E_n)^n}{\Delta t} + \frac{\partial A}{\partial t}E_n + Q\frac{\partial E_n{}^m}{\partial x} - \frac{\partial}{\partial x}\left(K_x A \frac{\partial E_n{}^m}{\partial x}\right) - \left[\frac{\partial A}{\partial t} - (S_R + S_R + S_1 + S_2 + S_I)\right]E_n{}^m \\ &= M_{E_n}{}^{as} + M_{E_n}{}^{rs} + M_{E_n}{}^{is} + M_{E_n}{}^{os1} + M_{E_n}{}^{os2} + AR_n \end{aligned} \tag{3.5.8.1.4}$$

According to Fully-implicit scheme, equation (3.5.8.1.4) can be separated into two equations as follows

$$\begin{aligned} &A\frac{(E_n)^{n+1/2} - (E_n)^n}{\Delta t} + \frac{\partial A}{\partial t}E_n + Q\frac{\partial E_n{}^m}{\partial x} - \frac{\partial}{\partial x}\left(K_x A \frac{\partial E_n{}^m}{\partial x}\right) - \left[\frac{\partial A}{\partial t} - (S_S + S_R + S_I + S_1 + S_2)\right]E_n{}^m = \\ &M_{E_n}{}^{as} + M_{E_n}{}^{rs} + M_{E_n}{}^{is} + M_{E_n}{}^{os1} + M_{E_n}{}^{os2} + AR_{E_n} \end{aligned} \tag{3.5.8.1.5}$$

$$\frac{(E_n)^{n+1}-(E_n)^{n+1/2}}{\Delta t}=0 \quad \textbf{(3.5.8.1.6)}$$

First, solve equation (3.5.8.1.5) and get $(E_n)^{n+1/2}$. Second, solve equation (3.5.8.1.6) together with algebraic equations for equilibrium reactions using BIOGEOCHEM scheme to obtain the individual species concentration. Iteration between these two steps is needed because reaction term in equation (3.5.8.1.5) needs to be updated by the results of (3.5.8.1.6).

To solve equation (3.5.8.1.5), assign

$$R_{HSn}=0 \quad and \quad L_{HSn}=(S_S+S_R+S_I+S_1+S_2)-\frac{\partial A}{\partial t} \quad \textbf{(3.5.8.1.7)}$$

Then the right hand side RHS_n and left hand side LHS_n should be continuously calculated same as that in section (3.5.7.1). Equation (3.5.8.1.5) is then simplified as

$$A\frac{(E_n)^{n+1/2}-(E_n)^n}{\Delta t}+\frac{\partial A}{\partial t}E_n+Q\frac{\partial E_n{}^m}{\partial x}-\frac{\partial}{\partial x}\left(K_xA\frac{\partial E_n{}^m}{\partial x}\right)+L_{HSn}E_n{}^m=R_{HSn}+AR_{E_n} \quad \textbf{(3.5.8.1.8)}$$

Express $E_n{}^m$ in terms of $(E_n{}^m/E_n)\,E_n{}^m$ to make E_n's as primary dependent variables,

$$A\frac{\partial E_n}{\partial t}+\frac{\partial A}{\partial t}E_n+Q\frac{\partial}{\partial x}\left(\frac{E_n{}^m}{E_n}E_n\right)-\frac{\partial}{\partial x}\left(K_xA\frac{E_n{}^m}{E_n}\frac{\partial E_n}{\partial x}\right)-\frac{\partial}{\partial x}\left(K_xA\frac{\partial(E_n{}^m/E_n)}{\partial x}E_n\right)+L_{HSn}\frac{E_n{}^m}{E_n}E_n=R_{HSn}+AR_{E_n} \quad \textbf{(3.5.8.1.9)}$$

Use Galerkin or Petrov-Galerkin FEM for the spatial discretization of transport equations. Integrate Equation (3.5.8.1.9) in the spatial dimensions over the entire region as follows.

$$\begin{aligned}&\int_{x_1}^{x_N}N_i\left[A\frac{\partial E_n}{\partial t}-\frac{\partial}{\partial x}\left(K_xA\frac{E_n{}^m}{E_n}\frac{\partial E_n}{\partial x}\right)\right]dx+\int_{x_1}^{x_N}W_i\left[Q\frac{\partial}{\partial x}\left(\frac{E_n{}^m}{E_n}E_n\right)-\frac{\partial}{\partial x}\left(K_xA\frac{\partial(E_n{}^m/E_n)}{\partial x}E_n\right)\right]dx\\&+\int_{x_1}^{x_N}N_i\left(L_{HSn}\frac{E_n{}^m}{E_n}+\frac{\partial A}{\partial t}\right)E_ndx=\int_{x_1}^{x_N}N_i\left(R_{HSn}+AR_n\right)dx\end{aligned} \quad \textbf{(3.5.8.1.10)}$$

Integrating by parts, we obtain

$$\begin{aligned}&\int_{x_1}^{x_N}N_iA\frac{\partial E_n}{\partial t}dx+\int_{x_1}^{x_N}W_iQ\frac{E_n{}^m}{E_n}\frac{\partial E_n}{\partial x}dx+\int_{x_1}^{x_N}W_iQ\frac{\partial(E_n{}^m/E_n)}{\partial x}E_ndx+\int_{x_1}^{x_N}\frac{dN_i}{dx}K_xA\frac{E_n{}^m}{E_n}\frac{\partial E_n}{\partial x}dx\\&+\int_{x_1}^{x_N}\frac{dW_i}{dx}K_xA\frac{\partial(E_n{}^m/E_n)}{\partial x}E_ndx+\int_{x_1}^{x_N}N_i\left(L_{HSn}\frac{E_n{}^m}{E_n}+\frac{\partial A}{\partial t}\right)E_ndx\\&=\int_{x_1}^{x_N}N_i\left(R_{HSn}+AR_{E_n}\right)dx+N_iK_xA\frac{E_n{}^m}{E_n}\frac{\partial E_n}{\partial x}\Bigg|_{B1}^{B2}+W_iK_xA\frac{\partial(E_n{}^m/E_n)}{\partial x}E_n\Bigg|_{B1}^{B2}\end{aligned} \quad \textbf{(3.5.8.1.11)}$$

Approximate solution E_n by a linear combination of the base functions as follows

$$E_n\approx\hat{E}_n=\sum_{j=1}^{N}E_{nj}(t)N_j(x) \quad \textbf{(3.5.8.1.12)}$$

Substituting Equation (3.5.8.1.12) into Equation (3.5.8.1.11), we obtain

$$\sum_{j=1}^{N}\left\{\left[\begin{array}{c}\int_{x_1}^{x_N} W_i Q \frac{E_n{}^m}{E_n}\frac{dN_j}{dx}dx + \int_{x_1}^{x_N} W_i Q \frac{\partial(E_n{}^m/E_n)}{\partial x} N_j dx + \int_{x_1}^{x_N}\frac{dW_i}{dx}K_x A\frac{\partial(E_n{}^m/E_n)}{\partial x}N_j dx \\ +\int_{x_1}^{x_N}\frac{dN_i}{dx}K_x A\frac{E_n{}^m}{E_n}\frac{dN_j}{dx}dx + \int_{x_1}^{x_N} N_i\left(L_{HSn}\frac{E_n{}^m}{E_n}+\frac{\partial A}{\partial t}\right)N_j dx\end{array}\right]E_{nj}(t)\right\}$$
$$+\sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N} N_i A N_j dx\right)\frac{\partial E_{nj}(t)}{\partial t}\right] = \int_{x_1}^{x_N} N_i\left(R_{HSn}+AR_{E_n}\right)dx + \sum n\left[N_i K_x A\frac{E_n{}^m}{E_n}\frac{\partial E_n}{\partial x} + W_i K_x A\frac{\partial(E_n{}^m/E_n)}{\partial x}E_n\right]_b \quad \textbf{(3.5.8.1.13)}$$

Equation (3.5.8.1.13) can be written in matrix form as

$$\left([L1]+[L2]+[L3]+[L4]+[L5]\right)\{E_n\}+[M]\left\{\frac{\partial E_n}{\partial t}\right\}=\{S\}+\{B\} \quad \textbf{(3.5.8.1.14)}$$

The matrices $[L1]$, $[L2]$, $[L3]$, $[L4]$, $[L5]$, $[M]$ and load vectors $\{S\}$, $\{B\}$ are given by

$$L1_{ij} = \int_{x_1}^{x_N} W_i Q\frac{E_n{}^m}{E_n}\frac{dN_j}{dx}dx \quad \textbf{(3.5.8.1.15)}$$

$$L2_{ij} = \int_{x_1}^{x_N} W_i Q\frac{\partial(E_n{}^m/E_n)}{\partial x}N_j dx \quad \textbf{(3.5.8.1.16)}$$

$$L3_{ij} = \int_{x_1}^{x_N}\frac{dW_i}{dx}K_x A\frac{\partial(E_n{}^m/E_n)}{\partial x}N_j dx \quad \textbf{(3.5.8.1.17)}$$

$$L4_{ij} = \int_{x_1}^{x_N}\frac{dN_i}{dx}K_x A\frac{E_n{}^m}{E_n}\frac{dN_j}{dx}dx \quad \textbf{(3.5.8.1.18)}$$

$$L5_{ij} = \int_{x_1}^{x_N} N_i\left(L_{HSn}\frac{E_n{}^m}{E_n}+\frac{\partial A}{\partial t}\right)N_j dx \quad \textbf{(3.5.8.1.19)}$$

$$M_{ij} = \int_{x_1}^{x_N} N_i A N_j dx \quad \textbf{(3.5.8.1.20)}$$

$$S_i = \int_{x_1}^{x_N} N_i\left(R_{HSn}+AR_{E_n}\right)dx \quad \textbf{(3.5.8.1.21)}$$

$$B_i = n\left[N_i K_x A\frac{E_n{}^m}{E_n}\frac{\partial E_n}{\partial x}+W_i K_x A\frac{\partial(E_n{}^m/E_n)}{\partial x}E_n\right]_b \quad \textbf{(3.5.8.1.22)}$$

Equation (3.5.8.1.14) is then simplified as

$$[L]\{E_n\}+[M]\left\{\frac{\partial E_n}{\partial t}\right\}=\{S\}+\{B\},\ \ where [L]=[L1]+[L2]+[L3]+[L4]+[L5] \quad \textbf{(3.5.8.1.23)}$$

Further,

$$[L]\left\{W_1 * E_n{}^{n+1/2}+W_2 * E_n{}^n\right\}+\frac{[M]}{\Delta t}\left\{E_n{}^{n+1/2}-E_n{}^n\right\}=\{S\}+\{B\} \quad \textbf{(3.5.8.1.24)}$$

So that

$$[CMATRX]\left\{E_n^{\ n+1/2}\right\} = \{RLD\} \quad \textbf{(3.5.8.1.25)}$$

where

$$[CMATRX] = \frac{[M]}{\Delta t} + W_1 * [L] \quad \textbf{(3.5.8.1.26)}$$

$$\{RLD\} = \left(\frac{[M]}{\Delta t} - W_2 * [L]\right)\left\{E_n^{\ n}\right\} + \left\{S\right\} + \left\{B\right\} \quad \textbf{(3.5.8.1.27)}$$

The above equations are used to solve for the kinetic variable concentration at interior nodes, where the boundary term {B} is zero.

At the junction nodes, assign

$$\{RLDW\} = \frac{[M]}{\Delta t} - W_2 * [L]\left\{E_n^{\ n}\right\} + \left\{S\right\} + \{nQE_n^{\ m}\} \quad \textbf{(3.5.8.1.28)}$$

Equation (3.5.8.1.25) is modified as

$$[CMATRX]\left\{E_n^{\ n+1/2}\right\} + Flux = \{RLDW\} \quad \textbf{(3.5.8.1.29)}$$

Junction concentration can be solved by the matrix equation assembled with equation (3.5.7.1.48), and (3.5.8.1.29).

For boundary node $i = b$, the boundary term $\{B\}$ should be continuously calculated as follows.

$$B_i = \left[N_i K_x A \frac{E_n^{\ m}}{E_n}\frac{\partial E_n}{\partial x} + W_i K_x A \frac{\partial (E_n^{\ m}/E_n)}{\partial x} E_n\right]_b = n\left[K_x A \frac{E_n^{\ m}}{E_n}\frac{\partial E_n}{\partial x} + K_x A \frac{\partial (E_n^{\ m}/E_n)}{\partial x} E_n\right]_b = n\left(K_x A \frac{\partial E_n^{\ m}}{\partial x}\right)_b \quad \textbf{(3.5.8.1.30)}$$

Dirichlet boundary condition

$$E_n^{\ m} = E_n^{\ m}(x_b, t) \quad \textbf{(3.5.8.1.31)}$$

Variable boundary condition

When flow is coming in from outside ($nQ < 0$)

$$n\left(QE_n^{\ m} - AK_x \frac{\partial E_n^{\ m}}{\partial x}\right) = nQE_n^{\ m}(x_b, t) \ \Rightarrow \ B_i = nQE_n^{\ m} - nQE_n^{\ m}(x_b, t) \quad \textbf{(3.5.8.1.32)}$$

When Flow is going out from inside ($nQ > 0$)

$$-\mathrm{nAK_x}\frac{\partial \mathrm{E_n^{\ m}}}{\partial \mathrm{x}} = 0 \Rightarrow \mathrm{B_i} = 0 \quad \textbf{(3.5.8.1.33)}$$

Cauchy boundary condition

$$n\left(QE_n^{\ m} - AK_x \frac{\partial E_n^{\ m}}{\partial x}\right) = Q_{En}(x_b, t) \ \Rightarrow \ B_i = nQE_n^{\ m} - Q_{En}(x_b, t) \quad \textbf{(3.5.8.1.34)}$$

Neumann boundary condition

$$-nAK_x \frac{\partial E_n{}^m}{\partial x} = Q_{En}(x_b, t) \quad \Rightarrow \quad B_i = -Q_{En}(x_b, t) \qquad \textbf{(3.5.8.1.35)}$$

3.5.8.2 Mixed Predictor-corrector/Operator-Splitting Scheme

Recall the continuity equation for kinetic-variables, equation (3.5.8.1.3), as follows.

$$\begin{aligned} & A\frac{\partial E_n}{\partial t} + \frac{\partial A}{\partial t}E_n + Q\frac{\partial E_n{}^m}{\partial x} - \frac{\partial}{\partial x}\left(K_x A \frac{\partial E_n{}^m}{\partial x}\right) - \left[\frac{\partial A}{\partial t} - (S_S + S_R + S_I + S_1 + S_2)\right]E_n{}^m = \\ & M_{E_n}{}^{as} + M_{E_n}{}^{rs} + M_{E_n}{}^{is} + M_{E_n}{}^{os1} + M_{E_n}{}^{os2} + AR_{E_n} \end{aligned} \qquad \textbf{(3.5.8.2.1)}$$

At *n+1*-th time step, equation (3.5.8.2.1) is approximated by

$$\begin{aligned} & A\frac{(E_n)^{n+1} - (E_n)^n}{\Delta t} + \frac{\partial A}{\partial t}E_n + Q\frac{\partial E_n{}^m}{\partial x} - \frac{\partial}{\partial x}\left(K_x A \frac{\partial E_n{}^m}{\partial x}\right) - \left[\frac{\partial A}{\partial t} - (S_S + S_R + S_1 + S_2 + S_I)\right]E_n{}^m \\ & = M_{E_n}{}^{as} + M_{E_n}{}^{rs} + M_{E_n}{}^{os1} + M_{E_n}{}^{os2} + M_{E_n}{}^{is} + AR_{E_n} \end{aligned} \qquad \textbf{(3.5.8.2.2)}$$

According to mixed predictor corrector/operator-splitting scheme, equation (3.5.8.2.2) can be separated into two equations as follows

$$\begin{aligned} & A\frac{(E_n^m)^{n+1} - (E_n^m)^n}{\Delta t} + \frac{\partial A}{\partial t}E_n^m + Q\frac{\partial E_n{}^m}{\partial x} - \frac{\partial}{\partial x}\left(K_x A \frac{\partial E_n{}^m}{\partial x}\right) - \left[\frac{\partial A}{\partial t} - (S_S + S_R + S_1 + S_2 + S_I)\right]E_n{}^m \\ & = M_{E_n}{}^{as} + M_{E_n}{}^{rs} + M_{E_n}{}^{os1} + M_{E_n}{}^{os2} + M_{E_n}{}^{is} + AR_{E_n}{}^n - \frac{\partial A}{\partial t}(E_n^{im})^n \end{aligned} \qquad \textbf{(3.5.8.2.3)}$$

$$\frac{E_n{}^{n+1} - [(E_n{}^m)^{n+1/2} + (E_n{}^{im})^n]}{\Delta t} = R_{E_n}{}^{n+1} - R_{E_n}{}^n - \frac{\partial(\ell nA)}{\partial t}(E_n^{im})^{n+1} + \frac{\partial(\ell nA)}{\partial t}(E_n^{im})^n \qquad \textbf{(3.5.8.2.4)}$$

First, solve equation (3.5.8.2.3) and get $(E_n^m)^{n+1/2}$. Second, solve equation (3.5.8.2.4) together with algebraic equations for equilibrium reactions using BIOGEOCHEM scheme to obtain $E_n{}^{n+1}$ and the individual species concentration.

To solve equation (3.5.8.2.3), assign and calculate R_{HSn} and L_{HSn} in the same way as that in Section (3.5.7.2). Equation (3.5.8.2.3) is then simplified as

$$A\frac{(E_n{}^m)^{n+1/2} - (E_n{}^m)^n}{\Delta t} + \frac{\partial A}{\partial t}E_n^m + Q\frac{\partial E_n{}^m}{\partial x} - \frac{\partial}{\partial x}\left(K_x A \frac{\partial E_n{}^m}{\partial x}\right) + L_{HS_n}E_n{}^m = R_{HS_n} + AR_{E_n}{}^n - \frac{\partial A}{\partial t}(E_n^{im})^n \qquad \textbf{(3.5.8.2.5)}$$

Use Galerkin or Petrov-Galerkin FEM for the spatial discretization of transport equations. For Galerkin method, choose weighting function identical to base functions. Integrate Equation (3.5.8.2.5) in the spatial dimensions over the entire region as follows.

$$\int_{x_1}^{x_N} N_i \left[A\frac{\partial E_n{}^m}{\partial t} - \frac{\partial}{\partial x}\left(K_x A \frac{\partial E_n{}^m}{\partial x} \right) \right] dx + \int_{x_1}^{x_N} W_i Q \frac{\partial E_n{}^m}{\partial x} dx + \int_{x_1}^{x_N} N_i \left(L_{HS_n} + \frac{\partial A}{\partial t} \right) E_n{}^m dx =$$
$$\int_{x_1}^{x_N} N_i \left[R_{HS_n} + A(R_{E_n})^n - \frac{\partial A}{\partial t}(E_n^{im})^n \right] dx \qquad \textbf{(3.5.8.2.6)}$$

Integrating by parts, we obtain

$$\int_{x_1}^{x_N} N_i A \frac{\partial E_n{}^m}{\partial t} dx + \int_{x_1}^{x_N} \frac{dN_i}{dx} K_x A \frac{\partial E_n{}^m}{\partial x} dx + \int_{x_1}^{x_N} W_i Q \frac{\partial E_n{}^m}{\partial x} dx + \int_{x_1}^{x_N} N_i \left(L_{HS_n} + \frac{\partial A}{\partial t} \right) E_n{}^m dx$$
$$= \int_{x_1}^{x_N} N_i \left[R_{HS_n} + A(R_{E_n})^n - \frac{\partial A}{\partial t}(E_n^{im})^n \right] dx + N_i K_x A \frac{\partial E_n{}^m}{\partial x} \Bigg|_{B1}^{B2} \qquad \textbf{(3.5.8.2.7)}$$

Approximate solution $E_n{}^m$ by a linear combination of the base functions as follows

$$E_n{}^m \approx \hat{E}_n{}^m = \sum_{j=1}^{N} E_{nj}{}^m(t) N_j(x) \qquad \textbf{(3.5.8.2.8)}$$

Substituting Equation (3.5.8.2.8) into Equation (3.5.8.2.7), we obtain

$$\sum_{j=1}^{N} \left[\left(\int_{x_1}^{x_N} W_i Q \frac{dN_j}{dx} dx + \int_{x_1}^{x_N} \frac{dN_i}{dx} K_x A \frac{dN_j}{dx} dx + \int_{x_1}^{x_N} N_i \left(L_{HS_n} + \frac{\partial A}{\partial t} \right) N_j dx \right) E_{nj}{}^m(t) \right]$$
$$+ \sum_{j=1}^{N} \left[\left(\int_{x_1}^{x_N} N_i A N_j dx \right) \frac{\partial E_{nj}{}^m(t)}{\partial t} \right] = \int_{x_1}^{x_N} N_i \left[R_{HS_n} + A(R_{E_n})^n - \frac{\partial A}{\partial t}(E_n^{im})^n \right] dx + \sum n \left(N_i K_x A \frac{\partial E_n{}^m}{\partial x} \right)_b \qquad \textbf{(3.5.8.2.9)}$$

Equation (3.5.8.2.9) can be written in matrix form as

$$\left([L1] + [L2] + [L3] \right) \left\{ E_n{}^m \right\} + [M] \left\{ \frac{\partial E_n{}^m}{\partial t} \right\} = \{S\} + \{B\} \qquad \textbf{(3.5.8.2.10)}$$

The matrices [$L1$], [$L2$], [$L3$], [M] and load vectors {S}, {B} are given by

$$L1_{ij} = \int_{x_1}^{x_N} W_i Q \frac{dN_j}{dx} dx \qquad \textbf{(3.5.8.2.11)}$$

$$L2_{ij} = \int_{x_1}^{x_N} \frac{dN_i}{dx} K_x A \frac{dN_j}{dx} dx \qquad \textbf{(3.5.8.2.12)}$$

$$L3_{ij} = \int_{x_1}^{x_N} N_i \left(L_{HS_n} + \frac{\partial A}{\partial t} \right) N_j dx \qquad \textbf{(3.5.8.2.13)}$$

$$M_{ij} = \int_{x_1}^{x_N} N_i A N_j dx \qquad \textbf{(3.5.8.2.14)}$$

$$S_i = \int_{x_1}^{x_N} N_i \left[R_{HS_n} + A(R_{E_n})^n - \frac{\partial A}{\partial t}(E_n^{im})^n \right] dx \qquad \textbf{(3.5.8.2.15)}$$

$$B_i = n\left(N_i K_x A \frac{\partial E_n{}^m}{\partial x} \right)_b \quad \textbf{(3.5.8.2.16)}$$

where all the terms listed above are calculated with the corresponding time weighting values. Equation (3.5.8.2.10) is then simplified as

$$[L]\left\{E_n{}^m\right\} + [M]\left\{\frac{\partial E_n{}^m}{\partial t}\right\} = \left\{S\right\} + \left\{B\right\}, \ \ where [L] = [L1] + [L2] + [L3] \quad \textbf{(3.5.8.2.17)}$$

Further,

$$[L]\left\{W_1 * (E_n{}^m)^{n+1/2} + W_2 * (E_n{}^m)^n\right\} + [M]\left\{\frac{(E_n{}^m)^{n+1/2} - (E_n{}^m)^n}{\Delta t}\right\} = \left\{S\right\} + \left\{B\right\} \quad \textbf{(3.5.8.2.18)}$$

So that

$$[CMATRX]\left\{(E_n{}^m)^{n+1/2}\right\} = \{RLD\} \quad \textbf{(3.5.8.2.19)}$$

where

$$[CMATRX] = \frac{[M]}{\Delta t} + W_1[L] \quad \textbf{(3.5.8.2.20)}$$

$$\{RLD\} = \left(\frac{[M]}{\Delta t} - W_2[L]\right)\left\{(E_n{}^m)^n\right\} + \left\{S\right\} + \left\{B\right\} \quad \textbf{(3.5.8.2.21)}$$

The above equations are used to solve for the kinetic variable concentration at interior nodes where boundary term {B} is zero.

For junction nodes, assign

$$\{RLDW\} = \frac{[M]}{\Delta t} - W_2[L]\left\{(E_n{}^m)^n\right\} + \left\{S\right\} + \{nQE_n{}^m\} \quad \textbf{(3.5.8.2.22)}$$

Equation (3.5.8.2.18) is modified as

$$[CMATRX]\left\{(E_n{}^m)^{n+1/2}\right\} + Flux = \{RLDW\} \quad \textbf{(3.5.8.2.23)}$$

Junction concentration can be solved by the matrix equation assembled with equation (3.5.7.2.32) and (3.5.8.2.23).

For boundary node $i = b$, the boundary term $\{B\}$ should be continuously calculated same as that using Fully-implicit scheme in section (3.5.8.1).

3.5.8.3 Operator-splitting

Recall the continuity equation for kinetic-variables, equation (3.5.8.1.3), as follows.

$$\begin{aligned} &A\frac{\partial E_n}{\partial t} + \frac{\partial A}{\partial t}E_n + Q\frac{\partial E_n{}^m}{\partial x} - \frac{\partial}{\partial x}\left(K_x A \frac{\partial E_n{}^m}{\partial x}\right) - \left[\frac{\partial A}{\partial t} - (S_S + S_R + S_I + S_1 + S_2)\right]E_n{}^m = \\ &M_{E_n}{}^{as} + M_{E_n}{}^{rs} + M_{E_n}{}^{is} + M_{E_n}{}^{os1} + M_{E_n}{}^{os2} + AR_{E_n} \end{aligned} \quad \textbf{(3.5.8.3.1)}$$

At $n+1$-th time step, equation (3.5.8.3.1) is approximated by

$$A\frac{(E_n)^{n+1}-(E_n)^n}{\Delta t}+\frac{\partial A}{\partial t}E_n+Q\frac{\partial E_n{}^m}{\partial x}-\frac{\partial}{\partial x}\left(K_xA\frac{\partial E_n{}^m}{\partial x}\right)-\left[\frac{\partial A}{\partial t}-(S_S+S_R+S_1+S_2+S_I)\right]E_n{}^m$$
$$=M_{E_n}{}^{as}+M_{E_n}{}^{rs}+M_{E_n}{}^{os1}+M_{E_n}{}^{os2}+M_{E_n}{}^{is}+AR_{E_n} \qquad \textbf{(3.5.8.3.2)}$$

According to Operator-splitting scheme, equation (3.5.8.3.2) can be separated into two equations as follows

$$A\frac{(E_n^m)^{n+1}-(E_n^m)^n}{\Delta t}+\frac{\partial A}{\partial t}E_n^m+Q\frac{\partial E_n{}^m}{\partial x}-\frac{\partial}{\partial x}\left(K_xA\frac{\partial E_n{}^m}{\partial x}\right)-\left[\frac{\partial A}{\partial t}-(S_S+S_R+S_1+S_2+S_I)\right]E_n{}^m$$
$$=M_{E_n}{}^{as}+M_{E_n}{}^{rs}+M_{E_n}{}^{os1}+M_{E_n}{}^{os2}+M_{E_n}{}^{is} \qquad \textbf{(3.5.8.3.3)}$$

$$\frac{E_n{}^{n+1}-[(E_n{}^m)^{n+1/2}+(E_n{}^{im})^n]}{\Delta t}=R_{E_n}{}^{n+1}-\frac{\partial(\ell nA)}{\partial t}(E_n^{im})^{n+1} \qquad \textbf{(3.5.8.3.4)}$$

First, solve equation (3.5.8.3.3) and get $(E_n^m)^{n+1/2}$. Second, solve equation (3.5.8.3.4) together with algebraic equations for equilibrium reactions using BIOGEOCHEM scheme to obtain $(E_n)^{n+1}$ and the individual species concentration.

To solve equation (3.5.8.3.3), assign and calculate R_{HSn} and L_{HSn} same as that in section (3.5.8.1). Equation (3.5.8.3.3) is then simplified as

$$A\frac{(E_n{}^m)^{n+1/2}-(E_n{}^m)^n}{\Delta t}+\frac{\partial A}{\partial t}E_n^m+Q\frac{\partial E_n{}^m}{\partial x}-\frac{\partial}{\partial x}\left(K_xA\frac{\partial E_n{}^m}{\partial x}\right)+L_{HS_n}E_n{}^m=R_{HS_n} \qquad \textbf{(3.5.8.3.5)}$$

Use Galerkin or Petrov-Galerkin FEM for the spatial discretization of transport equations. For Galerkin method, choose weighting function identical to base functions. Integrate Equation (3.5.8.3.5) in the spatial dimensions over the entire region as follows.

$$\int_{x_1}^{x_N}N_i\left[A\frac{\partial E_n{}^m}{\partial t}-\frac{\partial}{\partial x}\left(K_xA\frac{\partial E_n{}^m}{\partial x}\right)\right]dx+\int_{x_1}^{x_N}W_iQ\frac{\partial E_n{}^m}{\partial x}dx+\int_{x_1}^{x_N}N_i\left(L_{HS_n}+\frac{\partial A}{\partial t}\right)E_n{}^mdx=\int_{x_1}^{x_N}N_iR_{HS_n}dx \qquad \textbf{(3.5.8.3.6)}$$

Integrating by parts, we obtain

$$\int_{x_1}^{x_N}N_iA\frac{\partial E_n{}^m}{\partial t}dx+\int_{x_1}^{x_N}\frac{dN_i}{dx}K_xA\frac{\partial E_n{}^m}{\partial x}dx+\int_{x_1}^{x_N}W_iQ\frac{\partial E_n{}^m}{\partial x}dx+\int_{x_1}^{x_N}N_i\left(L_{HS_n}+\frac{\partial A}{\partial t}\right)E_n{}^mdx$$
$$=\int_{x_1}^{x_N}N_iR_{HS_n}dx+N_iK_xA\frac{\partial E_n{}^m}{\partial x}\bigg|_{B1}^{B2} \qquad \textbf{(3.5.8.3.7)}$$

Approximate solution $E_n{}^m$ by a linear combination of the base functions as follows

$$E_n{}^m\approx\hat{E}_n{}^m=\sum_{j=1}^{N}E_{nj}{}^m(t)N_j(x) \qquad \textbf{(3.5.8.3.8)}$$

Substituting Equation (3.5.8.3.8) into Equation (3.5.8.3.7), we obtain

$$\sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N} W_i Q\frac{dN_j}{dx}dx+\int_{x_1}^{x_N}\frac{dN_i}{dx}K_x A\frac{dN_j}{dx}dx+\int_{x_1}^{x_N}N_i\left(L_{HS_n}+\frac{\partial A}{\partial t}\right)N_j dx\right)E_{nj}^{\ m}(t)\right]$$
$$+\sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N}N_i A N_j dx\right)\frac{\partial E_{nj}^{\ m}(t)}{\partial t}\right]=\int_{x_1}^{x_N}N_i R_{HS_n}dx+\sum n\left(N_i K_x A\frac{\partial E_n^{\ m}}{\partial x}\right)_b \qquad \textbf{(3.5.8.3.9)}$$

Equation (3.5.8.3.9) can be written in matrix form as

$$([L1]+[L2]+[L3])\left\{E_n^{\ m}\right\}+[M]\left\{\frac{\partial E_n^{\ m}}{\partial t}\right\}=\left\{S\right\}+\left\{B\right\} \qquad \textbf{(3.5.8.3.10)}$$

The matrices $[L1]$, $[L2]$, $[L3]$, $[M]$ and load vectors $\{S\}$, $\{B\}$ are given by

$$L1_{ij}=\int_{x_1}^{x_N}W_i Q\frac{dN_j}{dx}dx \qquad \textbf{(3.5.8.3.11)}$$

$$L2_{ij}=\int_{x_1}^{x_N}\frac{dN_i}{dx}K_x A\frac{dN_j}{dx}dx \qquad \textbf{(3.5.8.3.12)}$$

$$L3_{ij}=\int_{x_1}^{x_N}N_i\left(L_{HS_n}+\frac{\partial A}{\partial t}\right)N_j dx \qquad \textbf{(3.5.8.3.13)}$$

$$M_{ij}=\int_{x_1}^{x_N}N_i A N_j dx \qquad \textbf{(3.5.8.3.14)}$$

$$S_i=\int_{x_1}^{x_N}N_i R_{HS_n}dx \qquad \textbf{(3.5.8.3.15)}$$

$$B_i=n\left(N_i K_x A\frac{\partial E_n^{\ m}}{\partial x}\right)_b \qquad \textbf{(3.5.8.3.16)}$$

where all the terms listed above are calculated with the corresponding time weighting values. Equation (3.5.8.3.10) is then simplified as

$$[L]\left\{E_n^{\ m}\right\}+[M]\left\{\frac{\partial E_n^{\ m}}{\partial t}\right\}=\left\{S\right\}+\left\{B\right\},\ \ where[L]=[L1]+[L2]+[L3] \qquad \textbf{(3.5.8.3.17)}$$

Further,

$$[L]\left\{W_1*(E_n^{\ m})^{n+1/2}+W_2*(E_n^{\ m})^n\right\}+[M]\left\{\frac{(E_n^{\ m})^{n+1/2}-(E_n^{\ m})^n}{\Delta t}\right\}=\left\{S\right\}+\left\{B\right\} \qquad \textbf{(3.5.8.3.18)}$$

So that

$$[CMATRX]\left\{(\mathrm{E_n}^{\ m})^{n+1/2}\right\}=\{RLD\} \qquad \textbf{(3.5.8.3.19)}$$

where

$$[CMATRX]=\frac{[M]}{\Delta t}+W_1[L] \qquad \textbf{(3.5.8.3.20)}$$

$$\{RLD\}=\left(\frac{[M]}{\Delta t}-W_2[L]\right)\left\{(E_n^{\ m})^n\right\}+\left\{S\right\}+\left\{B\right\} \qquad \textbf{(3.5.8.3.21)}$$

The above equations are used to solve for the kinetic variable concentration at interior nodes where boundary term $\{B\}$ is zero.

For junction nodes, assign

$$\{RLDW\} = \frac{[M]}{\Delta t} - W_2[L]\left\{(E_n{}^m)^n\right\} + \left\{S\right\} + \{nQE_n{}^m\} \quad \textbf{(3.5.8.3.22)}$$

Equation (3.5.8.3.18) is modified as

$$[CMATRX]\left\{(E_n{}^m)^{n+1/2}\right\} + Flux = \{RLDW\} \quad \textbf{(3.5.8.3.23)}$$

Junction concentration can be solved by the matrix equation assembled with equation (3.5.7.3.33) and (3.5.8.3.23).

For boundary node i = b, the boundary term {B} should be continuously calculated same as that using Fully-implicit scheme in section (3.5.8.1).

3.5.9 Application of the Modified Lagrangian-Eulerian Approach to the Largrangian Form of the Transport Equations

3.5.9.1 Fully-implicit scheme

The continuity equation for kinetic-variables in advective form at *(n+1)*-th time step, is shown as follows.

$$\begin{aligned} &A\frac{(E_n)^{n+1}-(E_n)^n}{\Delta t} + \frac{\partial A}{\partial t}E_n + Q\frac{\partial E_n{}^m}{\partial x} - \frac{\partial}{\partial x}\left(K_x A\frac{\partial E_n{}^m}{\partial x}\right) + \left[(S_S + S_R + S_1 + S_2 + S_I) - \frac{\partial A}{\partial t}\right]E_n{}^m \\ &= M_{E_n}{}^{as} + M_{E_n}{}^{rs} + M_{E_n}{}^{os1} + M_{E_n}{}^{os2} + M_{E_n}{}^{is} + AR_{E_n} \end{aligned} \quad \textbf{(3.5.9.1.1)}$$

[Option 1]

Express $E_n{}^m$ in terms of $E_n{}^m / E_n * E_n$ to make E_n's as primary dependent variables, equation (3.5.9.1.1) is modified as

$$\begin{aligned} &A\frac{(E_n)^{n+1}-(E_n)^n}{\Delta t} + \frac{\partial A}{\partial t}E_n + Q\frac{\partial \frac{E_n{}^m}{E_n}E_n}{\partial x} - \frac{\partial}{\partial x}\left(K_x A\frac{\partial \frac{E_n{}^m}{E_n}E_n}{\partial x}\right) + \\ &\left[(S_S + S_R + S_1 + S_2 + S_I) - \frac{\partial A}{\partial t}\right]\frac{E_n{}^m}{E_n}E_n = M_{E_n}{}^{as} + M_{E_n}{}^{rs} + M_{E_n}{}^{os1} + M_{E_n}{}^{os2} + M_{E_n}{}^{is} + AR_{E_n} \end{aligned} \quad \textbf{(3.5.9.1.2)}$$

According to Fully-implicit scheme, equation (3.5.9.1.2) can be separated into two equations as follows

$$A\frac{(E_n)^{n+1/2}-(E_n)^n}{\Delta t}+\frac{\partial A}{\partial t}E_n+Q\frac{\partial\frac{E_n^{\ m}}{E_n}E_n}{\partial x}-\frac{\partial}{\partial x}\left(K_xA\frac{\partial\frac{E_n^{\ m}}{E_n}E_n}{\partial x}\right)+$$

$$\left[(S_S+S_R+S_1+S_2+S_I)-\frac{\partial A}{\partial t}\right]\frac{E_n^{\ m}}{E_n}E_n=M_{E_n}^{\ as}+M_{E_n}^{\ rs}+M_{E_n}^{\ os1}+M_{E_n}^{\ os2}+M_{E_n}^{\ is}+AR_{E_n} \tag{3.5.9.1.3}$$

$$\frac{(E_n)^{n+1}-(E_n)^{n+1/2}}{\Delta t}=0 \tag{3.5.9.1.4}$$

First, solve equation (3.5.9.1.3) and get $(E_n)^{n+1/2}$. Second, solve equation (3.5.9.1.4) together with algebraic equations for equilibrium reactions using BIOGEOCHEM scheme to obtain the individual species concentration. Iteration between these two steps is needed because reaction term in equation (3.5.9.1.3) needs to be updated by the results of (3.5.9.1.4).

To solve equation (3.5.9.1.3), assign

$$R_{HS_n}=0 \quad and \quad L_{HS_n}=\left[(S_S+S_R+S_1+S_2+S_I)-\frac{\partial A}{\partial t}\right]\frac{E_n^{\ m}}{E_n} \tag{3.5.9.1.5}$$

Then the right hand side R_{HSn} and left hand side L_{HSn} should be continuously calculated as following.

$$M_{E_n}^{\ rs}=\begin{cases}S_R*E_{n^{rs}}, & if\ S_R>0 \ \Rightarrow\ R_{HSn}=R_{HSn}+M_{E_n}^{\ rs}\\ S_R*E_n^{\ m}, & if\ S_R\le 0 \ \Rightarrow\ L_{HSn}=L_{HSn}-S_R\end{cases} \tag{3.5.9.1.6}$$

$$M_{E_n}^{\ as}=\begin{cases}S_S*E_{n^{as}}, & if\ S_S>0 \ \Rightarrow\ R_{HSn}=R_{HSn}+M_{E_n}^{\ as},\\ S_S*E_n^{\ m}, & if\ S_S\le 0 \ \Rightarrow\ L_{HSn}=L_{HSn}-S_S\end{cases} \tag{3.5.9.1.7}$$

$$M_{E_n}^{\ os1}=\begin{cases}S_1*E_n^{\ m}{}_{os1}, & if\ S_1>0 \ \Rightarrow\ R_{HSn}=R_{HSn}+M_{E_n}^{\ os1}\\ S_1*E_n^{\ m}, & if\ S_1\le 0 \ \Rightarrow\ L_{HSn}=L_{HSn}-S_1\end{cases} \tag{3.5.9.1.8}$$

$$M_{E_n}^{\ os2}=\begin{cases}S_2*E_n^{\ m}{}_{os2}, & if\ S_2>0 \ \Rightarrow\ R_{HSn}=R_{HSn}+M_{E_n}^{\ os2}\\ S_2*E_n^{\ m}, & if\ S_2\le 0 \ \Rightarrow\ L_{HSn}=L_{HSn}-S_2\end{cases} \tag{3.5.9.1.9}$$

$$M_{E_n}^{\ is}=\begin{cases}S_I*E_n^{\ m}{}_{is}, & if\ S_I>0 \ \Rightarrow\ R_{HSn}=R_{HSn}+M_{E_n}^{\ is}\\ S_I*E_n^{\ m}, & if\ S_I\le 0 \ \Rightarrow\ L_{HSn}=L_{HSn}-S_I\end{cases} \tag{3.5.9.1.10}$$

Equation (3.5.9.1.3) is then simplified as

$$A\frac{(E_n)^{n+1/2}-(E_n)^n}{\Delta t}+\frac{\partial A}{\partial t}E_n+\left(Q\frac{E_n^{\ m}}{E_n}-K_xA\frac{\partial\frac{E_n^{\ m}}{E_n}}{\partial x}\right)\frac{\partial E_n}{\partial x}-\frac{\partial}{\partial x}\left(K_xA\frac{E_n^{\ m}}{E_n}\frac{\partial E_n}{\partial x}\right)+$$

$$\left[Q\frac{\partial\frac{E_n^{\ m}}{E_n}}{\partial x}-\frac{\partial}{\partial x}\left(K_xA\frac{\partial\frac{E_n^{\ m}}{E_n}}{\partial x}\right)+L_{HS_n}\right]E_n=R_{HS_n}+AR_{E_n} \tag{3.5.9.1.11}$$

Assign the true transport velocity V_{true} as follows.

$$AV_{true} = Q\frac{E_n^{\ m}}{E_n} - K_x A\frac{\partial}{\partial x}\left(\frac{E_n^{\ m}}{E_n}\right) \qquad \textbf{(3.5.9.1.12)}$$

$$K_{true} = K_x \frac{E_n^{\ m}}{E_n} \qquad \textbf{(3.5.9.1.13)}$$

$$L = Q\frac{\partial}{\partial x}\left(\frac{E_n^{\ m}}{E_n}\right) - \frac{\partial}{\partial x}\left[K_x A\frac{\partial}{\partial x}\left(\frac{E_n^{\ m}}{E_n}\right)\right] + L_{HS_n} \qquad \textbf{(3.5.9.1.14)}$$

Then equation (3.5.9.1.11) is simplified as

$$A\frac{(E_n)^{n+1/2} - (E_n)^n}{\Delta t} + AV_{true}\frac{\partial E_n}{\partial x} - \frac{\partial}{\partial x}\left(K_{true}A\frac{\partial E_n}{\partial x}\right) + \left(\frac{\partial A}{\partial t} + L\right)E_n = R_{HS_n} + AR_{E_n} \qquad \textbf{(3.5.9.1.15)}$$

Equation (13.5.7.1.15) in the Lagrangian and Eulerian form is as follows.

$$\frac{dE_n}{d\tau} = \frac{(E_n)^{n+1/2} - (E_n)^n}{\Delta t} + V_{true}\frac{\partial E_n}{\partial x} = 0 \qquad \textbf{(3.5.9.1.16)}$$

$$A\frac{dE_n}{d\tau} - \frac{\partial}{\partial x}\left(K_{true}A\frac{\partial E_n}{\partial x}\right) + \left(\frac{\partial A}{\partial t} + L\right)E_n = R_{HS_n} + AR_{E_n} \qquad \textbf{(3.5.9.1.17)}$$

First, solve equation (3.5.9.1.16) to obtain the Lagrangian values by particle tracking. Then, deal with Eulerian equation (3.5.9.1.17) by finite element method.

Equation (3.5.9.1.17) written in a slightly different form is shown as follows.

$$\frac{dE_n}{d\tau} - D + KE_n = R_L \qquad \textbf{(3.5.9.1.18)}$$

where

$$D = \frac{1}{A}\frac{\partial}{\partial x}\left(K_{true}A\frac{\partial E_n}{\partial x}\right) \qquad \textbf{(3.5.9.1.19)}$$

$$K = \frac{\left(\frac{\partial A}{\partial t} + L\right)}{A} \qquad \textbf{(3.5.9.1.20)}$$

$$R_L = \frac{R_{HS_n} + AR_{E_n}}{A} \qquad \textbf{(3.5.9.1.21)}$$

Equation (3.5.9.1.18) written in matrix form is then expressed as

$$\frac{[U]}{\Delta\tau}\left\{E_n^{\ n+1/2}\right\} - W_1\left\{D^{n+1}\right\} + W_1\left[K^{n+1}\right]\left\{E_n^{\ n+1/2}\right\} = \frac{[U]}{\Delta\tau}\left\{E_n^{\ *}\right\} + W_2\left\{D^*\right\} - W_2\left\{(KE_n)^*\right\} + W_1\left\{R_L^{\ n+1}\right\} + W_2\left\{R_L^{\ *}\right\} \qquad \textbf{(3.5.9.1.22)}$$

where $[K^{n+1}]$ is the diagonal matrix with K calculated at the $(n+1)$-th time step as its components, the diffusion term D expressed in term of E_n is solved by the following procedure.

Approximate D by a linear combination of the base functions as follows.

$$D \approx \hat{D} = \sum_{j=1}^{N} D_j(t) N_j(x) \qquad \textbf{(3.5.9.1.23)}$$

According to equation (3.5.9.1.19), the integration of equation (3.5.9.1.22) can be written as

$$\int_{x_1}^{x_N} N_i A D dx = \int_{x_1}^{x_N} N_i A \sum_{j=1}^{N} D_j(t) N_j(x) dx = \int_{x_1}^{x_N} N_i \frac{\partial}{\partial x}\left(K_{true} A \frac{\partial E_n}{\partial x} \right) dx \qquad \textbf{(3.5.9.1.24)}$$

Integrating by parts, we obtain

$$\sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N} N_i A N_j dx\right) D_j\right] = -\int_{x_1}^{x_N} \frac{dN_i}{dx}(K_{true}A)\frac{\partial E_n}{\partial x} dx + N_i K_{true} A \frac{\partial E_n}{\partial x}\bigg|_{B1}^{B2} \qquad \textbf{(3.5.9.1.25)}$$

Approximate E_n by a linear combination of the base functions as follows.

$$E_n \approx \hat{E}_n = \sum_{j=1}^{N} E_{nj}(t) N_j(x) \qquad \textbf{(3.5.9.1.26)}$$

Equation (3.5.9.1.25) is further expressed as

$$\sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N} N_i A N_j dx\right) D_j\right] = -\sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N} \frac{dN_i}{dx}(K_{true}A)\frac{dN_j}{\partial x} dx\right)(E_n)_j\right] + N_i K_{true} A \frac{\partial E_n}{\partial x}\bigg|_{B1}^{B2} \qquad \textbf{(3.5.9.1.27)}$$

Assign matrices $[A1]$ and $[A2]$ and load vector $\{B1\}$ as following

$$A1_{ij} = \int_{x_1}^{x_N} N_i A N_j dx \qquad \textbf{(3.5.9.1.28)}$$

$$A2_{ij} = \int_{x_1}^{x_N} \frac{dN_i}{dx}(K_{true}A)\frac{dN_j}{dx} dx \qquad \textbf{(3.5.9.1.29)}$$

$$B1_i = \left(n N_i K_{true} A \frac{\partial E_n}{\partial x} \right)_b \qquad \textbf{(3.5.9.1.30)}$$

Equation (3.5.9.1.27) is expressed as

$$[A1]\{D\} = -[A2]\{E_n\} + \{B1\} \qquad \textbf{(3.5.9.1.31)}$$

Lump matrix $[A1]$ into diagonal matrix and assign

$$QE_{ij} = A2_{ij} / A1_{ii} \qquad \textbf{(3.5.9.1.32)}$$

$$B_i = B1_i / A1_{ii} \qquad \textbf{(3.5.9.1.33)}$$

Then

$$\{D\} = -[QE]\{E_n\} + \{B\} \qquad \textbf{(3.5.9.1.34)}$$

where boundary term $\{B\}$ is calculated as follows

$$B_i = \left(nN_i K_x A \frac{\partial E_n{}^m}{\partial x} \right)_b \Big/ A1_{ii} - \left[nN_i K_x A \frac{\partial}{\partial x} \left(\frac{E_n{}^m}{E_n} \right) E_n \right]_b \Big/ A1_{ii} \quad \textbf{(3.5.9.1.35)}$$

Dirichlet boundary condition

$$\begin{aligned} & E_n{}^m = E_n{}^m (x_b, t) \quad \Rightarrow \\ & B_i = nN_i K_x A \frac{(E_n{}^m)_j - E_n{}^m (x_b, t)}{\Delta x} \Big/ A1_{ii} - nN_i K_x A \frac{(E_n{}^m / E_n)_j - (E_n{}^m / E_n)_i}{\Delta x} \left(E_n \right)_i \Big/ A1_{ii} \end{aligned} \quad \textbf{(3.5.9.1.36)}$$

where j is the interior node connected to the boundary node.

Variable boundary condition

When flow is coming in from outside (nQ < 0)

$$\begin{aligned} & n \left(QE_n{}^m - AK_x \frac{\partial E_n{}^m}{\partial x} \right) = nQE_n{}^m (x_b, t) \quad \Rightarrow \\ & B_i = \left[nQE_n{}^m - nQE_n{}^m (x_b, t) \right] \Big/ A1_{ii} - nN_i K_x A \frac{(E_n{}^m / E_n)_j - (E_n{}^m / E_n)_i}{\Delta x} \left(E_n \right)_i \Big/ A1_{ii} \end{aligned} \quad \textbf{(3.5.9.1.37)}$$

where j is the interior node connected to the boundary node.

When Flow is going out from inside (nQ > 0)

$$-nAK_x \frac{\partial E_n{}^m}{\partial x} = 0 \quad \Rightarrow \quad B_i = -nN_i K_x A \frac{(E_n{}^m / E_n)_j - (E_n{}^m / E_n)_i}{\Delta x} \left(E_n \right)_i \Big/ A1_{ii} \quad \textbf{(3.5.9.1.38)}$$

where j is the interior node connected to the boundary node.

Cauchy boundary condition

$$\begin{aligned} & n \left(QE_n{}^m - AK_x \frac{\partial E_n{}^m}{\partial x} \right) = Q_{En} (x_b, t) \quad \Rightarrow \\ & B_i = \left[nQE_n{}^m - Q_{En} (x_b, t) \right] \Big/ A1_{ii} - nN_i K_x A \frac{(E_n{}^m / E_n)_j - (E_n{}^m / E_n)_i}{\Delta x} \left(E_n \right)_i \Big/ A1_{ii} \end{aligned} \quad \textbf{(3.5.9.1.39)}$$

where j is the interior node connected to the boundary node.

Neumann boundary condition

$$-nAK_x \frac{\partial E_n{}^m}{\partial x} = Q_{En} (x_b, t) \quad \Rightarrow \quad B_i = -Q_{En} (x_b, t) \ - nN_i K_x A \frac{(E_n{}^m / E_n)_j - (E_n{}^m / E_n)_i}{\Delta x} \left(E_n \right)_i \Big/ A1_{ii} \quad \textbf{(3.5.9.1.40)}$$

where j is the interior node connected to the boundary node.

Equation (3.5.9.1.22) can be written as matrix equation as following

$$\frac{[U]}{\Delta\tau}\left\{E_n^{\ n+1/2}\right\}+W_1[QE^{n+1}]\left\{E_n^{\ n+1/2}\right\}+W_1\left[K^{n+1}\right]\left\{E_n^{\ n+1/2}\right\}$$
$$=\frac{[U]}{\Delta\tau}\left\{E_n^{\ *}\right\}-W_2\left\{\left(KE_n\right)^*\right\}+W_2\left\{D^*\right\}+W_1\left\{R_L^{\ n+1}\right\}+W_2\left\{R_L^{\ *}\right\}+W_1\left\{B^{n+1}\right\} \quad \textbf{(3.5.9.1.41)}$$

[Option 2]

Express E_n^m in terms of E_n-E_n^m and E_n^m/E_n*E_n to make E_n's as primary dependent variables, equation (3.5.9.1.1) is modified as

$$A\frac{(E_n)^{n+1}-(E_n)^n}{\Delta t}+\frac{\partial A}{\partial t}E_n+Q\frac{\partial E_n}{\partial x}-\frac{\partial}{\partial x}\left(K_xA\frac{\partial E_n}{\partial x}\right)+\left[(S_S+S_R+S_1+S_2+S_I)-\frac{\partial A}{\partial t}\right]\frac{E_n^{\ m}}{E_n}E_n$$
$$=\left[Q\frac{\partial E_n^{\ im}}{\partial x}-\frac{\partial}{\partial x}\left(K_xA\frac{\partial E_n^{\ im}}{\partial x}\right)\right]+M_{E_n}^{\ \ as}+M_{E_n}^{\ \ rs}+M_{E_n}^{\ \ os1}+M_{E_n}^{\ \ os2}+M_{E_n}^{\ \ is}+AR_{E_n} \quad \textbf{(3.5.9.1.42)}$$

According to Fully-implicit scheme, equation (3.5.9.1.42) can be separated into two equations as follows

$$A\frac{(E_n)^{n+1/2}-(E_n)^n}{\Delta t}+\frac{\partial A}{\partial t}E_n+Q\frac{\partial E_n}{\partial x}-\frac{\partial}{\partial x}\left(K_xA\frac{\partial E_n}{\partial x}\right)+\left[(S_S+S_R+S_1+S_2+S_I)-\frac{\partial A}{\partial t}\right]\frac{E_n^{\ m}}{E_n}E_n$$
$$=\left[Q\frac{\partial E_n^{\ im}}{\partial x}-\frac{\partial}{\partial x}\left(K_xA\frac{\partial E_n^{\ im}}{\partial x}\right)\right]+M_{E_n}^{\ \ as}+M_{E_n}^{\ \ rs}+M_{E_n}^{\ \ os1}+M_{E_n}^{\ \ os2}+M_{E_n}^{\ \ is}+AR_{E_n} \quad \textbf{(3.5.9.1.43)}$$

$$\frac{(E_n)^{n+1}-(E_n)^{n+1/2}}{\Delta t}=0 \quad \textbf{(3.5.9.1.44)}$$

First, solve equation (3.5.9.1.43) and get $E_n^{\ n+1/2}$. Second, solve equation (3.5.9.1.44) together with algebraic equations for equilibrium reactions using BIOGEOCHEM scheme to obtain the individual species concentration and $(E_n)^{n+1}$. Iteration between these two steps is needed because reaction term in equation (3.5.9.1.43) needs to be updated by the results of (3.5.9.1.44).

To solve equation (3.5.9.1.43), assign

$$R_{HS_n}=0 \quad and \quad L_{HS_n}=\left[(S_S+S_R+S_1+S_2+S_I)-\frac{\partial A}{\partial t}\right]\frac{E_n^{\ m}}{E_n} \quad \textbf{(3.5.9.1.45)}$$

Then the right hand side R_{HSn} and left hand side L_{HSn} should be continuously calculated as following.

$$M_{E_n}^{\ \ rs}=\begin{cases}S_R*E_{n^{rs}}, \ \ if\ S_R>0 \ \Rightarrow\ R_{HS_n}=R_{HS_n}+M_{E_n}^{\ \ rs}\\ S_R*E_n^{\ m}, \ \ if\ S_R\le 0 \ \Rightarrow\ L_{HS_n}=L_{HS_n}-S_R*E_n^{\ m}/E_n\end{cases} \quad \textbf{(3.5.9.1.46)}$$

$$M_{E_n}^{\ \ as}=\begin{cases}S_S*E_{n^{as}}, \ \ if\ S_S>0 \ \Rightarrow\ R_{HS_n}=R_{HS_n}+M_{E_n}^{\ \ as},\\ S_S*E_n^{\ m}, \ \ if\ S_S\le 0 \ \Rightarrow\ LHS_n=L_{HS_n}-S_S*E_n^{\ m}/E_n\end{cases} \quad \textbf{(3.5.9.1.47)}$$

$$M_{E_n}^{\ \ os1}=\begin{cases}S_1*E_{n^{os1}}^m, \ \ if\ S_1>0 \ \Rightarrow\ R_{HS_n}=R_{HS_n}+M_{E_n}^{\ \ os1}\\ S_1*E_n^{\ m}, \ \ if\ S_1\le 0 \ \Rightarrow\ L_{HS_n}=L_{HS_n}-S_1*E_n^{\ m}/E_n\end{cases} \quad \textbf{(3.5.9.1.48)}$$

$$M_{E_n}{}^{os2} = \begin{cases} S_2 * E_{n\ os2}^{m}, & if\ S_2 > 0 \Rightarrow R_{HS_n} = R_{HS_n} + M_{E_n}{}^{os2} \\ S_2 * E_n{}^m, & if\ S_2 \le 0 \Rightarrow L_{HS_n} = L_{HS_n} - S_2 * E_n{}^m / E_n \end{cases} \quad \textbf{(3.5.9.1.49)}$$

$$M_{E_n}{}^{is} = \begin{cases} S_I * E_{n\ is}^{m}, & if\ S_I > 0 \Rightarrow R_{HS_n} = R_{HS_n} + M_{E_n}{}^{is} \\ S_I * E_n{}^m, & if\ S_I \le 0 \Rightarrow L_{HS_n} = L_{HS_n} - S_I * E_n{}^m / E_n \end{cases} \quad \textbf{(3.5.9.1.50)}$$

Equation (3.5.9.1.43) is then simplified as

$$A\frac{(E_n)^{n+1/2} - (E_n)^n}{\Delta t} + \frac{\partial A}{\partial t}E_n + Q\frac{\partial E_n}{\partial x} - \frac{\partial}{\partial x}\left(K_x A \frac{\partial E_n}{\partial x}\right) + L_{HS_n}E_n = \left[Q\frac{\partial E_n{}^{im}}{\partial x} - \frac{\partial}{\partial x}\left(K_x A\frac{\partial E_n{}^{im}}{\partial x}\right)\right] + R_{HS_n} + AR_{E_n} \quad \textbf{(3.5.9.1.51)}$$

Assign the true transport velocity V_{true} as follows.

$$AV_{true} = Q \quad \textbf{(3.5.9.1.52)}$$

Then equation (3.5.9.1.51) is simplified as

$$A\frac{(E_n)^{n+1/2} - (E_n)^n}{\Delta t} + A^n V_{true}\frac{\partial E_n}{\partial x} - \frac{\partial}{\partial x}\left(K_x A \frac{\partial E_n}{\partial x}\right) + \left(L_{HS_n} + \frac{\partial A}{\partial t}\right)E_n = \left[Q\frac{\partial E_n{}^{im}}{\partial x} - \frac{\partial}{\partial x}\left(K_x A\frac{\partial E_n{}^{im}}{\partial x}\right)\right] + R_{HS_n} + AR_{E_n} \quad \textbf{(3.5.9.1.53)}$$

Equation (13.5.9.1.53) in the Lagrangian and Eulerian form is as follows.

$$\frac{dE_n}{d\tau} = \frac{(E_n)^{n+1/2} - (E_n)^n}{\Delta t} + V_{true}\frac{\partial E_n}{\partial x} = 0 \quad \textbf{(3.5.9.1.54)}$$

$$A\frac{dE_n}{d\tau} - \frac{\partial}{\partial x}\left(K_x A \frac{\partial E_n}{\partial x}\right) + \left(L_{HS_n} + \frac{\partial A}{\partial t}\right)E_n = \left[Q\frac{\partial E_n{}^{im}}{\partial x} - \frac{\partial}{\partial x}\left(K_x A\frac{\partial E_n{}^{im}}{\partial x}\right)\right] + R_{HSn} + AR_{E_n} \quad \textbf{(3.5.9.1.55)}$$

First, solve equation (3.5.9.1.54) to obtain the Lagrangian values by particle tracking. Then, deal with Eulerian equation (3.5.9.1.55) by finite element method.

Equation (3.5.9.1.55) written in a slightly different form is shown as follows.

$$\frac{dE_n}{d\tau} - D + K * E_n = T + R_L \quad \textbf{(3.5.9.1.56)}$$

where

$$D = \frac{1}{A}\frac{\partial}{\partial x}\left(K_x A\frac{\partial E_n}{\partial x}\right) \quad \textbf{(3.5.9.1.57)}$$

$$K = \frac{\left(L_{HS_n} + \frac{\partial A}{\partial t}\right)}{A} \quad \textbf{(3.5.9.1.58)}$$

$$R_L = \frac{R_{HS_n} + AR_{E_n}}{A} \quad \textbf{(3.5.9.1.59)}$$

$$T = \frac{1}{A}\left[Q\frac{\partial E_n{}^{im}}{\partial x} - \frac{\partial}{\partial x}\left(K_x A\frac{\partial E_n{}^{im}}{\partial x}\right)\right] \quad \textbf{(3.5.9.1.60)}$$

Equation (3.5.9.1.56) written in matrix form is then expressed as

$$\frac{[U]}{\Delta\tau}\left\{E_n^{\ n+1/2}\right\}-W_1\left\{D^{n+1}\right\}+W_1\left[K^{n+1}\right]\left\{E_n^{\ n+1/2}\right\}=$$
$$\frac{[U]}{\Delta\tau}\left\{E_n^{\ *}\right\}+W_2\left\{D^*\right\}-W_2\left\{(KE_n)^*\right\}+W_1\left\{T^{n+1}\right\}+W_2\left\{T^*\right\}+W_1\left\{R_L^{\ n+1}\right\}+W_2\left\{R_L^{\ *}\right\} \quad \textbf{(3.5.9.1.61)}$$

where $[K^{n+1}]$ is the diagonal matrix with K calculated at $(n+1)$-th time step as its components, the diffusion term D expressed in term of E_n and term T expressed in term of $E_n^{\ im}$ is solved by the following procedure.

Approximate D by a linear combination of the base functions as follows.

$$D\approx\hat{D}=\sum_{j=1}^{N}D_j(t)N_j(x) \quad \textbf{(3.5.9.1.62)}$$

According to equation (3.5.9.1.57), the integration of equation (3.5.9.1.62) can be written as

$$\int_{x_1}^{x_N}N_i ADdx=\int_{x_1}^{x_N}N_i A\sum_{j=1}^{N}D_j(t)N_j(x)dx=\int_{x_1}^{x_N}N_i\frac{\partial}{\partial x}\left(K_x A\frac{\partial E_n}{\partial x}\right)dx \quad \textbf{(3.5.9.1.63)}$$

Integrating by parts, we obtain

$$\sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N}N_i AN_j dx\right)D_j\right]=-\int_{x_1}^{x_N}\frac{dN_i}{dx}(K_x A)\frac{\partial E_n}{\partial x}dx+N_i K_x A\frac{\partial E_n}{\partial x}\bigg|_{B1}^{B2} \quad \textbf{(3.5.9.1.64)}$$

Approximate E_n by a linear combination of the base functions as follows.

$$E_n\approx\hat{E}_n=\sum_{j=1}^{N}E_{nj}(t)N_j(x) \quad \textbf{(3.5.9.1.65)}$$

Equation (3.5.9.1.64) is further expressed as

$$\sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N}N_i AN_j dx\right)D_j\right]=-\sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N}\frac{dN_i}{dx}(K_x A)\frac{dN_j}{\partial x}dx\right)(E_n)_j\right]+N_i K_x A\frac{\partial E_n}{\partial x}\bigg|_{B1}^{B2} \quad \textbf{(3.5.9.1.66)}$$

Assign matrices [$A1$] and [$A2$] and load vector { $B1$ } as following

$$A1_{ij}=\int_{x_1}^{x_N}N_i AN_j dx \quad \textbf{(3.5.9.1.67)}$$

$$A2_{ij}=\int_{x_1}^{x_N}\frac{dN_i}{dx}(K_x A)\frac{dN_j}{dx}dx \quad \textbf{(3.5.9.1.68)}$$

$$B1_i=\left(nN_i K_x A\frac{\partial E_n}{\partial x}\right)_b \quad \textbf{(3.5.9.1.69)}$$

Equation (3.5.9.1.66) is expressed as

$$[A1]\{D\}=-[A2]\{E_n\}+\{B1\} \quad \textbf{(3.5.9.1.70)}$$

Lump matrix [$A1$] into diagonal matrix and assign

$$QE_{ij} = A2_{ij} / A1_{ii} \tag{3.5.9.1.71}$$

$$QB1_i = B1_i / A1_{ii} \tag{3.5.9.1.72}$$

Then

$$\{D\} = -[QE]\{E_n\} + \{QB1\} \tag{3.5.9.1.73}$$

Approximate T by a linear combination of the base functions as follows.

$$T \approx \hat{T} = \sum_{j=1}^{N} T_j(t) N_j(x) \tag{3.5.9.1.74}$$

According to equation (3.5.9.1.60), the integration of equation (3.5.9.1.74) can be written as

$$\int_{x_1}^{x_N} N_i ATdx = \int_{x_1}^{x_N} N_i A \sum_{j=1}^{N} T_j(t) N_j(x) dx = \int_{x_1}^{x_N} N_i \left[Q\frac{\partial E_n{}^{im}}{\partial x} - \frac{\partial}{\partial x}\left(K_x A \frac{\partial E_n{}^{im}}{\partial x}\right)\right] dx \tag{3.5.9.1.75}$$

Integrating by parts, we obtain

$$\sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N} N_i AN_j dx\right) T_j\right] = \int_{x_1}^{x_N} N_i Q \frac{\partial E_n{}^{im}}{\partial x} dx + \int_{x_1}^{x_N} \frac{dN_i}{dx} K_x A \frac{\partial E_n{}^{im}}{\partial x} dx + N_i K_x A \frac{\partial E_n{}^{im}}{\partial x}\bigg|_{B1}^{B2} \tag{3.5.9.1.76}$$

Approximate $E_n{}^{im}$ by a linear combination of the base functions as follows.

$$E_n{}^{im} \approx \hat{E}_n{}^{im} = \sum_{j=1}^{N} E_{nj}{}^{im}(t) N_j(x) \tag{3.5.9.1.77}$$

Equation (3.5.9.1.76) is further expressed as

$$\begin{aligned}\sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N} N_i AN_j dx\right) T_j\right] &= \sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N} N_i Q \frac{dN_j}{dx} dx\right)(E_n{}^{im})_j\right] \\ &+ \sum_{j=1}^{N}\left[\left(\int_{x_1}^{x_N} \frac{dN_i}{dx} K_x A \frac{dN_j}{dx} dx\right)(E_n{}^{im})_j\right] + N_i K_x A \frac{\partial E_n{}^{im}}{\partial x}\bigg|_{B1}^{B2}\end{aligned} \tag{3.5.9.1.78}$$

Assign matrices [$A3$], and load vector {$B2$} as following

$$A3_{ij} = \int_{x_1}^{x_N} N_i Q \frac{dN_j}{dx} dx \tag{3.5.9.1.79}$$

$$B2_i = \left(-nN_i K_x A \frac{\partial E_n{}^{im}}{\partial x}\right)_b \tag{3.5.9.1.80}$$

Assign

$$QT_{ij} = (A2_{ij} + A3_{ij}) / A1_{ii} \tag{3.5.9.1.81}$$

$$QB2_i = B2_i / A1_{ii} \tag{3.5.9.1.82}$$

Equation (3.5.9.1.78) is expressed as

$$\{T\} = [QT]\{E_n^{im}\} + \{QB2\} \quad \textbf{(3.5.9.1.83)}$$

So that

$$\{D\} + \{T\} = -[QE]\{E_n\} + [QT]\{E_n^{im}\} + \{B\} \quad \textbf{(3.5.9.1.84)}$$

where boundary term $\{B\}$ is calculated as follows

$$B_i = QB1_i + QB2_i = \left(nK_x A \frac{\partial E_n^m}{\partial x} \right)_b \bigg/ A1_{ii} \quad \textbf{(3.5.9.1.85)}$$

Dirichlet boundary condition

$$E_n^m = E_n^m(x_b, t) \ \Rightarrow \ B_i = nN_i K_x A \frac{(E_n^m)_j - E_n^m(x_b, t)}{\Delta x} \bigg/ A1_{ii} \quad \textbf{(3.5.9.1.86)}$$

where j is the interior node connected to the boundary node.

Variable boundary condition

When flow is coming in from outside ($nQ < 0$)

$$n\left(QE_n^m - AK_x \frac{\partial E_n^m}{\partial x} \right) = nQE_n^m(x_b, t) \ \Rightarrow \ B_i = \left[nQE_n^m - nQE_n^m(x_b, t) \right] \big/ A1_{ii} \quad \textbf{(3.5.9.1.87)}$$

When Flow is going out from inside (nQ > 0)

$$-nAK_x \frac{\partial E_n^m}{\partial x} = 0 \ \Rightarrow \ B_i = 0 \quad \textbf{(3.5.9.1.88)}$$

Cauchy boundary condition

$$n\left(QE_n^m - AK_x \frac{\partial E_n^m}{\partial x} \right) = Q_{En}(x_b, t) \ \Rightarrow \ B_i = \left[nQE_n^m - Q_{En}(x_b, t) \right] \big/ A1_{ii} \quad \textbf{(3.5.9.1.89)}$$

Neumann boundary condition

$$-nAK_x \frac{\partial E_n^m}{\partial x} = Q_{En}(x_b, t) \ \Rightarrow \ B_i = -Q_{En}(x_b, t) \quad \textbf{(3.5.9.1.90)}$$

Equation (3.5.9.1.61) can be written as matrix equation as following

$$\begin{aligned} &\frac{[U]}{\Delta\tau}\{E_n^{n+1/2}\} + W_1[QE^{n+1}]\{E_n^{n+1/2}\} + W_1\left[K^{n+1}\right]\{E_n^{n+1/2}\} - W_1[QT^{n+1}]\left\{\left(E_n^{im}\right)^{n+1/2}\right\} \\ &= \frac{[U]}{\Delta\tau}\{E_n^*\} - W_2\left\{(KE_n)^*\right\} + W_2\left(\{D^*\} + \{T^*\}\right) + W_1\{R_L^{n+1}\} + W_2\{R_L^*\} + W_1\{B^{n+1}\} \end{aligned} \quad \textbf{(3.5.9.1.91)}$$

So that

$$[CMATRX]\{E_n^{n+1/2}\} = \{RLD\} \quad \textbf{(3.5.9.1.92)}$$

where

$$[CMATRX] = \frac{[U]}{\Delta\tau} + W_1[QE^{n+1}] + W_1\left[K^{n+1}\right] - W_1[QT^{n+1}\frac{E_n^{im}}{E_n}] \quad \textbf{(3.5.9.1.93)}$$

$$\{RLD\} = \frac{[U]}{\Delta\tau}\left\{E_n^*\right\} - W_2\left\{(KE_n)^*\right\} + W_2\left(\left\{D^*\right\} + \left\{T^*\right\}\right) + W_1\left\{R_L^{\ n+1}\right\} + W_2\left\{R_L^{\ *}\right\} + W_1\left\{B^{n+1}\right\} \quad \textbf{(3.5.9.1.94)}$$

At junctions, if nQ > 0, flow is going from reach to the junction. Assign

$$\left\{RLDW\right\} = \left\{RLD\right\} + \left\{nQE_n^{\ m} / A1_{ii}^{\ n+1}\right\} - W_1\left\{B^{n+1}\right\} - W_2\left\{\left(nK_x A\frac{\partial E_n^{\ m}}{\partial x}\right)^n \Big/ A1_{ii}^{\ n+1}\right\} \quad \textbf{(3.5.9.1.95)}$$

Equation (3.5.9.1.89) is modified as

$$[CMATRX]\left\{\mathrm{E_n}^{n+1/2}\right\} + Flux / A1_{ii} = \{RLDW\} \quad \textbf{(3.5.9.1.96)}$$

If nQ < 0, flow is going from junction to the reach, apply equation (3.5.7.1.57),

$$Flux_i = n\left[Q(E_n^{\ m})_i - K_x A\frac{(E_n^{\ m})_j - (E_n^{\ m})_i}{\Delta x}\right] \quad \textbf{(3.5.9.1.97)}$$

So that junction concentration and flux can be solved by the matrix equation assembled with equation (3.5.7.1.48), (3.5.9.1.96) and (3.5.9.1.97).

3.5.9.2 Mixed Predictor-corrector/Operator-Splitting Scheme

The continuity equation for kinetic-variables in advective form is shown as follows.

$$\begin{aligned} &A\frac{\partial E_n}{\partial t} + \frac{\partial A}{\partial t}E_n + Q\frac{\partial E_n^{\ m}}{\partial x} - \frac{\partial}{\partial x}\left(K_x A\frac{\partial E_n^{\ m}}{\partial x}\right) + \left[(S_S + S_R + S_1 + S_2 + S_I) - \frac{\partial A}{\partial t}\right]E_n^{\ m} \\ &= M_{E_n}^{\ as} + M_{E_n}^{\ rs} + M_{E_n}^{\ os1} + M_{E_n}^{\ os2} + M_{E_n}^{\ is} + AR_{E_n} \end{aligned} \quad \textbf{(3.5.9.2.1)}$$

At (*n+1*)-th time step, equation (3.5.9.2.1) is approximated by

$$\begin{aligned} &A\frac{(E_n)^{n+1} - (E_n)^n}{\Delta t} + \frac{\partial A}{\partial t}E_n + Q\frac{\partial E_n^{\ m}}{\partial x} - \frac{\partial}{\partial x}\left(K_x A\frac{\partial E_n^{\ m}}{\partial x}\right) + \left[(S_S + S_R + S_1 + S_2 + S_I) - \frac{\partial A}{\partial t}\right]E_n^{\ m} \\ &= M_{E_n}^{\ as} + M_{E_n}^{\ rs} + M_{E_n}^{\ os1} + M_{E_n}^{\ os2} + M_{E_n}^{\ is} + AR_{E_n} \end{aligned} \quad \textbf{(3.5.9.2.2)}$$

According to Mixed Predictor-corrector/Operator-Splitting Scheme, equation (3.5.9.2.2) can be separated into two equations as follows

$$\begin{aligned} &A\frac{(E_n^m)^{n+1/2} - (E_n^m)^n}{\Delta t} + \frac{\partial A}{\partial t}E_n^m + Q\frac{\partial E_n^{\ m}}{\partial x} - \frac{\partial}{\partial x}\left(K_x A\frac{\partial E_n^{\ m}}{\partial x}\right) + \left[(S_S + S_R + S_1 + S_2 + S_I) - \frac{\partial A}{\partial t}\right]E_n^{\ m} \\ &= M_{E_n}^{\ as} + M_{E_n}^{\ rs} + M_{E_n}^{\ os1} + M_{E_n}^{\ os2} + M_{E_n}^{\ is} + AR_{E_n}^{\ \ n} - \frac{\partial A}{\partial t}(E_n^{im})^n \end{aligned} \quad \textbf{(3.5.9.2.3)}$$

$$\frac{E_n^{\ n+1} - [(E_n^{\ m})^{n+1/2} + (E_n^{\ im})^n]}{\Delta t} = R_{E_n}^{\ \ n+1} - R_{E_n}^{\ \ n} - \frac{\partial(\ell nA)}{\partial t}(E_n^{im})^{n+1} + \frac{\partial(\ell nA)}{\partial t}(E_n^{im})^n \quad \textbf{(3.5.9.2.4)}$$

First, solve equation (3.5.9.2.3) and get $(E_n{}^m)^{n+1/2}$. Second, solve equation (3.5.9.2.4) together with algebraic equations for equilibrium reactions using BIOGEOCHEM scheme to obtain the individual species concentration.

To solve equation (3.5.9.2.3), assign and calculate R_{HSn} and L_{HSn} the same as that in section (3.5.7.2). Equation (3.5.9.2.3) is then simplified as

$$A\frac{(E_n{}^m)^{n+1/2}-(E_n{}^m)^n}{\Delta t}+\frac{\partial A}{\partial t}E_n^m+Q\frac{\partial E_n{}^m}{\partial x}-\frac{\partial}{\partial x}\left(K_x A\frac{\partial E_n{}^m}{\partial x}\right)+L_{HS_n}E_n{}^m=R_{HS_n}+AR_n{}^n-\frac{\partial A}{\partial t}(E_n^{im})^n \tag{3.5.9.2.5}$$

Assign the true transport velocity V_{true} as follows.

$$AV_{true}=Q \tag{3.5.9.2.6}$$

Then equation (3.5.9.2.5) is simplified as

$$A\frac{(E_n{}^m)^{n+1/2}-(E_n{}^m)^n}{\Delta t}+AV_{true}\frac{\partial E_n{}^m}{\partial x}-\frac{\partial}{\partial x}\left(K_x A\frac{\partial E_n{}^m}{\partial x}\right)+\left(L_{HS_n}+\frac{\partial A}{\partial t}\right)E_n{}^m=R_{HS_n}+AR_{E_n}{}^n-\frac{\partial A}{\partial t}(E_n^{im})^n \tag{3.5.9.2.7}$$

Equation (3.5.9.2.7) in the Lagrangian and Eulerian form is as follows.

$$\frac{dE_n{}^m}{d\tau}=\frac{(E_n{}^m)^{n+1/2}-(E_n{}^m)^n}{\Delta t}+V_{true}\frac{\partial E_n{}^m}{\partial x}=0 \tag{3.5.9.2.8}$$

$$A\frac{dE_n{}^m}{d\tau}-\frac{\partial}{\partial x}\left(K_x A\frac{\partial E_n{}^m}{\partial x}\right)+\left(L_{HS_n}+\frac{\partial A}{\partial t}\right)E_n{}^m=R_{HS_n}+AR_{E_n}{}^n-\frac{\partial A}{\partial t}(E_n^{im})^n \tag{3.5.9.2.9}$$

First, solve equation (3.5.9.2.8) to obtain the Lagrangian values by particle tracking. Then, deal with Eulerian equation (3.5.9.2.9) by finite element method.

Equation (3.5.9.2.9) written in a slightly different form is shown as follows.

$$\frac{dE_n{}^m}{d\tau}-D+K*E_n{}^m=R_L \tag{3.5.9.2.10}$$

where

$$D=\frac{1}{A}\frac{\partial}{\partial x}\left(K_x A\frac{\partial E_n{}^m}{\partial x}\right) \tag{3.5.9.2.11}$$

$$K=\frac{\left(L_{HS_n}+\frac{\partial A}{\partial t}\right)}{A} \tag{3.5.9.2.12}$$

$$R_L=\frac{R_{HS_n}+AR_{E_n}{}^n-\frac{\partial A}{\partial t}(E_n^{im})^n}{A} \tag{3.5.9.2.13}$$

Equation (3.5.9.2.10) written in matrix form is then expressed as

$$\frac{[U]}{\Delta\tau}\left\{\left(E_n{}^m\right)^{n+1/2}\right\} - W_1\left\{D^{n+1}\right\} + W_1\left[K^{n+1}\right]\left\{\left(E_n{}^m\right)^{n+1/2}\right\} = \frac{[U]}{\Delta\tau}\left\{\left(E_n{}^m\right)^*\right\} + W_2\left\{D^*\right\} - W_2\left\{\left(KE_n{}^m\right)^*\right\} + W_1\left\{R_L{}^{n+1}\right\} + W_2\left\{R_L{}^*\right\} \quad \textbf{(3.5.9.2.14)}$$

According to section 3.5.9.1,

$$\left\{D\right\} = -[QE]\left\{E_n{}^m\right\} + \left\{B\right\} \quad \textbf{(3.5.9.2.15)}$$

where *[QE*] and {*B*} are the same as those in section 3.5.9.1.

Equation (3.5.9.2.14) can be written as matrix equation as following

$$\frac{[U]}{\Delta\tau}\left\{\left(E_n{}^m\right)^{n+1/2}\right\} + W_1[QE^{n+1}]\left\{\left(E_n{}^m\right)^{n+1/2}\right\} + W_1\left[K^{n+1}\right]\left\{\left(E_n{}^m\right)^{n+1/2}\right\} = \frac{[U]}{\Delta\tau}\left\{\left(E_n{}^m\right)^*\right\} - W_2\left\{\left(KE_n{}^m\right)^*\right\} + W_2\left\{D^*\right\} + W_1\left\{R_L{}^{n+1}\right\} + W_2\left\{R_L{}^*\right\} + W_1\left\{B^{n+1}\right\} \quad \textbf{(3.5.9.2.16)}$$

So that

$$[CMATRX]\left\{\left(E_n{}^m\right)^{n+1/2}\right\} = \{RLD\} \quad \textbf{(3.5.9.2.17)}$$

where

$$[CMATRX] = \frac{[U]}{\Delta\tau} + W_1[QE^{n+1}] + W_1\left[K^{n+1}\right] \quad \textbf{(3.5.9.2.18)}$$

$$\{RLD\} = \frac{[U]}{\Delta\tau}\left\{\left(E_n{}^m\right)^*\right\} - W_2\left\{\left(KE_n{}^m\right)^*\right\} + W_2\left\{D^*\right\} + W_1\left\{R_L{}^{n+1}\right\} + W_2\left\{R_L{}^*\right\} + W_1\left\{B^{n+1}\right\} \quad \textbf{(3.5.9.2.19)}$$

At junctions, if $nQ > 0$, flow is going from reach to the junction. Assign

$$\left\{RLDW\right\} = \left\{RLD\right\} + \left\{nQE_n{}^m / A1_{ii}{}^{n+1}\right\} - W_1\left\{B^{n+1}\right\} - W_2\left\{\left(nK_x A\frac{\partial E_n{}^m}{\partial x}\right)^n \Big/ A1_{ii}{}^{n+1}\right\} \quad \textbf{(3.5.9.2.20)}$$

Equation (3.5.9.1.17) is modified as

$$[CMATRX]\left\{(E_n{}^m)^{n+1/2}\right\} + Flux/A1_{ii} = \{RLDW\} \quad \textbf{(3.5.9.2.21)}$$

If nQ < 0, flow is going from junction to the reach, apply equation (3.5.7.1.37),

$$Flux_i = n\left[Q(E_n{}^m)_i - K_x A\frac{(E_n{}^m)_j - (E_n{}^m)_i}{\Delta x}\right] \quad \textbf{(3.5.9.2.22)}$$

Junction concentration can be solved by the matrix equation assembled with equation (3.5.7.2.32), (3.5.9.2.21) and (3.5.9.2.22).

3.5.9.3 Operator-Splitting

The continuity equation for kinetic-variables in advective form is shown as follows.

$$A\frac{\partial E_n}{\partial t}+\frac{\partial A}{\partial t}E_n+Q\frac{\partial E_n{}^m}{\partial x}-\frac{\partial}{\partial x}\left(K_xA\frac{\partial E_n{}^m}{\partial x}\right)+\left[(S_S+S_R+S_1+S_2+S_I)-\frac{\partial A}{\partial t}\right]E_n{}^m$$
$$=M_{E_n}{}^{as}+M_{E_n}{}^{rs}+M_{E_n}{}^{os1}+M_{E_n}{}^{os2}+M_{E_n}{}^{is}+AR_{E_n} \quad \textbf{(3.5.9.3.1)}$$

At n+1-th time step, equation (3.5.9.3.1) is approximated by

$$A\frac{(E_n)^{n+1}-(E_n)^n}{\Delta t}+\frac{\partial A}{\partial t}E_n+Q\frac{\partial E_n{}^m}{\partial x}-\frac{\partial}{\partial x}\left(K_xA\frac{\partial E_n{}^m}{\partial x}\right)+\left[(S_S+S_R+S_1+S_2+S_I)-\frac{\partial A}{\partial t}\right]E_n{}^m$$
$$=M_{E_n}{}^{as}+M_{E_n}{}^{rs}+M_{E_n}{}^{os1}+M_{E_n}{}^{os2}+M_{E_n}{}^{is}+AR_{E_n} \quad \textbf{(3.5.9.3.2)}$$

According to Operator-splitting scheme, equation (3.5.9.3.2) can be separated into two equations as follows

$$A\frac{(E_n{}^m)^{n+1/2}-(E_n{}^m)^n}{\Delta t}+\frac{\partial A}{\partial t}E_n^m+Q\frac{\partial E_n{}^m}{\partial x}-\frac{\partial}{\partial x}\left(K_xA\frac{\partial E_n{}^m}{\partial x}\right)+\left[(S_S+S_R+S_1+S_2+S_I)-\frac{\partial A}{\partial t}\right]E_n{}^m$$
$$=M_{E_n}{}^{as}+M_{E_n}{}^{rs}+M_{E_n}{}^{os1}+M_{E_n}{}^{os2}+M_{E_n}{}^{is} \quad \textbf{(3.5.9.3.3)}$$

$$\frac{(E_n)^{n+1}-[(E_n{}^m)^{n+1/2}+(E_n{}^{im})^n]}{\Delta t}=AR_{E_n}{}^{n+1}-\frac{\partial(\ell nA)}{\partial t}(E_n^{im})^{n+1} \quad \textbf{(3.5.9.3.4)}$$

First, solve equation (3.5.9.3.3) and get $(E_n{}^m)^{n+1/2}$. Second, solve equation (3.5.9.3.4) together with algebraic equations for equilibrium reactions using BIOGEOCHEM scheme to obtain the individual species concentration.

To solve equation (3.5.9.3.3), assign and calculate R_{HSn} and L_{HSn} the same as that in section (3.5.8.1). Equation (3.5.9.3.3) is then simplified as

$$A\frac{(E_n{}^m)^{n+1/2}-(E_n{}^m)^n}{\Delta t}+\frac{\partial A}{\partial t}E_n^m+Q\frac{\partial E_n{}^m}{\partial x}-\frac{\partial}{\partial x}\left(K_xA\frac{\partial E_n{}^m}{\partial x}\right)+L_{HS_n}E_n{}^m=R_{HS_n} \quad \textbf{(3.5.9.3.5)}$$

Assign the true transport velocity V_{true} as follows.

$$AV_{true}=Q \quad \textbf{(3.5.9.3.6)}$$

Then equation (3.5.9.3.5) is simplified as

$$A\frac{(E_n{}^m)^{n+1/2}-(E_n{}^m)^n}{\Delta t}+AV_{true}\frac{\partial E_n{}^m}{\partial x}-\frac{\partial}{\partial x}\left(K_xA\frac{\partial E_n{}^m}{\partial x}\right)+\left(L_{HS_n}+\frac{\partial A}{\partial t}\right)E_n{}^m=R_{HS_n} \quad \textbf{(3.5.9.3.7)}$$

Equation (3.5.9.3.7) in the Lagrangian and Eulerian form is as follows.

$$\frac{dE_n{}^m}{d\tau}=\frac{(E_n{}^m)^{n+1/2}-(E_n{}^m)^n}{\Delta t}+V_{true}\frac{\partial E_n{}^m}{\partial x}=0 \quad \textbf{(3.5.9.3.8)}$$

$$A\frac{dE_n{}^m}{d\tau}-\frac{\partial}{\partial x}\left(K_xA\frac{\partial E_n{}^m}{\partial x}\right)+\left(L_{HS_n}+\frac{\partial A}{\partial t}\right)E_n{}^m=R_{HS_n} \quad \textbf{(3.5.9.3.9)}$$

First, solve equation (3.5.9.3.8) to obtain the lagrangian values by particle tracking. Then, deal with Eulerian equation (3.5.9.3.9) by finite element method.

Equation (3.5.9.3.9) written in a slightly different form is shown as follows.

$$\frac{dE_n^{\ m}}{d\tau} - D + K * E_n^{\ m} = R_L \tag{3.5.9.3.10}$$

where

$$D = \frac{1}{A}\frac{\partial}{\partial x}\left(K_x A \frac{\partial E_n^{\ m}}{\partial x}\right) \tag{3.5.9.3.11}$$

$$K = \frac{\left(L_{HS_n} + \frac{\partial A}{\partial t}\right)}{A} \tag{3.5.9.3.12}$$

$$R_L = \frac{R_{HS_n}}{A} \tag{3.5.9.3.13}$$

Equation (3.5.9.3.10) written in matrix form is then expressed as

$$\begin{aligned}&\frac{[U]}{\Delta\tau}\left\{\left(E_n^{\ m}\right)^{n+1/2}\right\} - W_1\left\{D^{n+1}\right\} + W_1\left[K^{n+1}\right]\left\{\left(E_n^{\ m}\right)^{n+1/2}\right\} = \\ &\frac{[U]}{\Delta\tau}\left\{\left(E_n^{\ m}\right)^*\right\} + W_2\left\{D^*\right\} - W_2\left\{\left(KE_n^{\ m}\right)^*\right\} + W_1\left\{R_L^{\ n+1}\right\} + W_2\left\{R_L^{\ *}\right\}\end{aligned} \tag{3.5.9.3.14}$$

According to section 3.5.9.1,

$$\{D\} = -[QE]\left\{E_n^{\ m}\right\} + \{B\} \tag{3.5.9.3.15}$$

where $[QE]$ and $\{B\}$ are the same as those in section 3.5.9.1.

Equation (3.5.9.3.14) can be written as matrix equation as following

$$\begin{aligned}&\frac{[U]}{\Delta\tau}\left\{\left(E_n^{\ m}\right)^{n+1/2}\right\} + W_1[QE^{n+1}]\left\{\left(E_n^{\ m}\right)^{n+1/2}\right\} + W_1\left[K^{n+1}\right]\left\{\left(E_n^{\ m}\right)^{n+1/2}\right\} \\ &= \frac{[U]}{\Delta\tau}\left\{\left(E_n^{\ m}\right)^*\right\} - W_2\left\{\left(KE_n^{\ m}\right)^*\right\} + W_2\left\{D^*\right\} + W_1\left\{R_L^{\ n+1}\right\} + W_2\left\{R_L^{\ *}\right\} + W_1\left\{B^{n+1}\right\}\end{aligned} \tag{3.5.9.3.16}$$

So that

$$[CMATRX]\left\{\left(E_n^{\ m}\right)^{n+1/2}\right\} = \{RLD\} \tag{3.5.9.3.17}$$

where

$$[CMATRX] = \frac{[U]}{\Delta\tau} + W_1[\mathrm{QE}^{\mathrm{n+1}}] + W_1\left[K^{n+1}\right] \tag{3.5.9.3.18}$$

$$\{RLD\} = \frac{[U]}{\Delta\tau}\left\{\left(E_n^{\ m}\right)^*\right\} - W_2\left\{\left(KE_n^{\ m}\right)^*\right\} + W_2\left\{D^*\right\} + W_1\left\{R_L^{\ n+1}\right\} + W_2\left\{R_L^{\ *}\right\} + W_1\left\{\mathrm{B}^{\mathrm{n+1}}\right\} \tag{3.5.9.3.19}$$

At junctions, if $nQ > 0$, flow is going from reach to the junction. Assign

$$\{RLDW\} = \{RLD\} + \left\{nQE_n^{\ m} / A1_{ii}^{\ n+1}\right\} - W_1\left\{B^{n+1}\right\} - W_2\left\{\left(nK_x A \frac{\partial E_n^{\ m}}{\partial x}\right)^n \Big/ A1_{ii}^{\ n+1}\right\} \tag{3.5.9.3.20}$$

Equation (3.5.9.1.19) is modified as

$$[CMATRX]\left\{(E_n{}^m)^{n+1/2}\right\} + Flux/A1_{ii} = \{RLDW\} \quad \textbf{(3.5.9.3.21)}$$

If $nQ < 0$, flow is going from junction to the reach, apply equation (3.5.7.1.37),

$$Flux_i = n\left[Q(E_n{}^m)_i - K_x A\frac{(E_n{}^m)_j - (E_n{}^m)_i}{\Delta x}\right] \quad \textbf{(3.5.9.3.22)}$$

Junction concentration can be solved by the matrix equation assembled with equation (3.5.7.3.32), (3.5.9.3.21) and (3.5.9.3.22).

3.5.10 Application of the Lagrangian-Eulerian Approach for All Interior Nodes and Downstream Boundary Nodes with the Finite Element Method Applied to the Conservative Form of the Transport Equations for the Upstream Flux Boundaries to Solve 1-D Kinetic Variable Transport

3.5.10.1 Fully-Implicit Scheme

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.5.9.1, and the matrix equation for junction and upstream boundary nodes is obtained through the same procedure as that in section 3.5.7.1.

3.5.10.2 Mixed Predictor-Corrector and Operator-Splitting Method

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.5.9.2, and the matrix equation for junction and upstream boundary nodes is obtained through the same procedure as that in section 3.5.7.2.

3.5.10.3 Operator-Splitting Approach

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.5.9.3, and the matrix equation for junction and upstream boundary nodes is obtained through the same procedure as that in section 3.5.7.3.

3.5.11 Application of the Lagrangian-Eulerian Approach for All Interior Nodes and Downstream Boundary Nodes with the Finite Element Method Applied to the Advective Form of the Transport Equations for the Upstream Flux Boundaries to Solve 1-D Kinetic Variable Transport

3.5.11.1 Fully-Implicit Scheme

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.5.9.1, and the matrix equation for junction and upstream boundary nodes is obtained through the same procedure as that in section 3.5.8.1.

3.5.11.2 Mixed Predictor-Corrector and Operator-Splitting Method

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.5.9.2, and the matrix equation for junction and upstream boundary nodes is obtained through the same procedure as that in section 3.5.8.2.

3.5.11.3 Operator-Splitting Approach

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.5.9.3, and the matrix equation for junction and upstream boundary nodes is obtained through the same procedure as that in section 3.5.8.3

3.6 Solving Two-Dimensional Overland Water Quality Transport Equations

In this section, we present the numerical approaches employed to solve the governing equations of reactive chemical transport. Ideally, one would like to use a numerical approach that is accurate, efficient, and robust. Depending on the specific problem at hand, different numerical approaches may be more suitable. For research applications, accuracy is a primary requirement, because one does not want to distort physics due to numerical errors. On the other hand, for large field-scale problems, efficiency and robustness are primary concerns as long as accuracy remains within the bounds of uncertainty associated with model parameters. Thus, to provide accuracy for research applications and efficiency and robustness for practical applications, three coupling strategies were investigated to deal with reactive chemistry. They are: (1) a fully-implicit scheme, (2) a mixed predictor-corrector/operator-splitting method, and (3) an operator-splitting method. For each time-tep, we first solve the advective-dispersive transport equation with or without reaction terms, kinetic-variable by kinetic-variable. We then solve the reactive chemical system node-by-node to yield concentrations of all species.

Five numerical options are provided to solve the advective-dispersive transport equations: Option 1- application of the Finite Element Method (FEM) to the conservative form of the transport equations, Option 2 - application of the FEM to the advective form of the transport equations, Option 3 - application of the modified Lagrangian-Eulerian (LE) approach to the Largrangian form of the transport equations, Option 4 - LE approach for all interior nodes and downstream boundary nodes with the FEM applied to the conservative form of the transport equations for the upstream flux boundaries, and Option 5 - LE approach for all interior and downstream boundary nodes with the FEM applied to the advective form of the transport equations for upstream flux boundaries.

3.6.1 Two-Dimensional Bed Sediment Balance Equation

At n+1-th time step, the continuity equation for 2-D bed sediment transport, equation (3.2.1), is approximated as

$$\frac{M_n^{\ n+1} - M_n^{\ n}}{\Delta t} \approx W_1(D_n^{\ n+1} - R_n^{\ n+1}) + W_2(D_n^{\ n} - R_n^{\ n}) \tag{3.6.1.1}$$

So that

$$M_n^{\ n+1} = M_n^{\ n} + W_1\left(D_n^{\ n+1} - R_n^{\ n+1}\right)\Delta t + W_2\left(D_n^{\ n} - R_n^{\ n}\right)\Delta t \tag{3.6.1.2}$$

If the calculated $M_n^{\ n+1} < 0$, assign $M_n^{\ n+1} = 0$, so that

$$R_n^{\ n+1} \approx \left(M_n^{\ n} - M_n^{\ n+1}\right)/\left(W_1 \Delta t\right) + W_2\left(D_n^{\ n} - R_n^{\ n}\right)/W_1 + D_n^{\ n+1} \tag{3.6.1.3}$$

3.6.2 Application of the Finite Element Method to the Conservative Form of the Transport Equations to Solve 2-D Suspended Sediment Transport

Recall the governing equation for 2-D suspended sediment transport, equation (2.6.10), as follows

$$\frac{\partial(hS_n)}{\partial t} + \nabla\cdot(\mathbf{q}S_n) - \nabla\cdot(h\mathbf{K}\cdot\nabla S_n) = M_{S_n^{as}} + M_{S_n^{rs}} + R_n - D_n,\ n\in[1, N_s] \tag{3.6.2.1}$$

Assign and calculate the right hand side term R_{HS} and left hand side term L_{HS} as follows.

$$\begin{array}{l} \textit{Assign}\ \ L_{HS} = 0\ \ \textit{and}\ \ R_{HS} = R_n - D_n \qquad \textit{then continuously calculate} \\ (1):\ \ \textit{If}\ \ S_S \le 0,\ \ L_{HS} = L_{HS} - S_S,\ \ \textit{ELSE}\ \ \ R_{HS} = R_{HS} + S_S * S_n^{as} \\ (2):\ \ \textit{If}\ \ S_R \le 0,\ \ L_{HS} = L_{HS} - S_R,\ \ \textit{ELSE}\ \ \ R_{HS} = R_{HS} + S_R * S_n^{rs} \end{array} \tag{3.6.2.2}$$

where S_n^{as} is the concentration of the *n*-th fraction suspended sediment in the artificial source and S_n^{rs} is the concentration of the *n*-th fraction suspended sediment in the rainfall source. Then equation (3.6.2.1) is modified as

$$\frac{\partial(hS_n)}{\partial t} + \nabla\cdot(\mathbf{q}S_n) - \nabla\cdot(h\mathbf{K}\cdot\nabla S_n) + L_{HS} * S_n = R_{HS} \tag{3.6.2.3}$$

Use Galerkin or Petrov-Galerkin finite-element method for the spatial descretization of transport equation: choose weighting function identical to base function. For Petriov-Galerkin method, apply weighting function one-order higher than the base function to advection term. Integrate equation (3.6.2.3) in the spatial dimensions over the entire region as follows.

$$\int_R N_i\left[\frac{\partial(hS_n)}{\partial t} - \nabla\cdot(h\mathbf{K}\cdot\nabla S_n) + L_{HS} * S_n\right]dR + \int_R W_i\nabla\cdot(\mathbf{q}S_n)\ dR = \int_R N_i R_{HS}\ dR \tag{3.6.2.4}$$

Further, we obtain

$$\begin{aligned} &\int_R N_i \frac{\partial(hS_n)}{\partial t}\ dR - \int_R \nabla W_i\cdot\mathbf{q}S_n\ dR + \int_R \nabla N_i\cdot(h\mathbf{K}\cdot\nabla S_n)\ dR + \int_R N_i\ L_{HS} * S_n dR \\ &= \int_R N_i\ R_{HS} dR - \int_B \mathbf{n}\cdot W_i\mathbf{q}S_n\ dB + \int_B \mathbf{n}\cdot N_i(h\mathbf{K}\cdot\nabla S_n)\ dB \end{aligned} \tag{3.6.2.5}$$

Approximate solution S_n by a linear combination of the base functions as shown by equation (3.6.2.6).

$$S_n \approx \hat{S}_n = \sum_{j=1}^{N} S_{nj}(t) N_j(R) \tag{3.6.2.6}$$

Substituting equation (3.6.2.6) into equation (3.6.2.5), we obtain

$$\begin{aligned} &\sum_{j=1}^{N}\left\{\left[\int_R N_i(\frac{\partial h}{\partial t}+L_{HS})N_j\ dR-\int_R \nabla W_i\cdot \mathbf{q}N_j\ dR+\int_R \nabla N_i\cdot h\mathbf{K}\cdot\nabla N_j\ dR\right]S_{nj}(t)\right\} \\ &+\sum_{j=1}^{N}\left[\left(\int_R N_i h N_j\ dR\right)\frac{dS_{nj}(t)}{dt}\right]=\int_R N_i R_{HS}\ dR-\int_B \mathbf{n}\cdot(W_i\mathbf{q}S_n-N_i h\mathbf{K}\cdot\nabla S_n)dB \end{aligned} \tag{3.6.2.7}$$

Equation (3.6.2.7) can be written in matrix form as

$$[CMATRX1]\left\{\frac{dS_n}{dt}\right\}+([Q1]+[Q2]+[Q3])\{S_n\}=\{SS\}+\{B\} \tag{3.6.2.8}$$

where the matrices [CMATRX1], [Q1], [Q2], [Q3] and load vectors {RLD}, and {B} are given by

$$CMATRX1_{ij}=\int_R N_i h N_j dR \tag{3.6.2.9}$$

$$Q1_{ij}=\int_R N_i(\frac{\partial h}{\partial t}+L_{HS})N_j dR \tag{3.6.2.10}$$

$$Q2_{ij}=-\int_R \nabla W_i\cdot \mathbf{q}N_j dR \tag{3.6.2.11}$$

$$Q3_{ij}=-\int_R \nabla N_i\cdot h\mathbf{K}\cdot\nabla N_j dR \tag{3.6.2.12}$$

$$SS_{ij}=\int_R N_i R_{HS} dR \tag{3.6.2.13}$$

$$B_i=-\int_B n\cdot(W_i\mathbf{q}S_n-N_i h\mathbf{K}\cdot\nabla S_n)dB \tag{3.6.2.14}$$

where all the integrations are evaluated with the corresponding time weighting values.

At n+1-th time step, equation (3.6.2.8) is approximated as

$$[CMATRX1]\left\{\frac{S_n{}^{n+1}-S_n{}^{n}}{\Delta t}\right\}+[CMATRX2]\{W_1 S_n{}^{n+1}+W_2 S_n{}^{n}\}=\{SS\}+\{B\} \tag{3.6.2.15}$$

where

$$[CMATRX2]=[Q1]+[Q2]+[Q3] \tag{3.6.2.16}$$

So that

$$[CMATRX]\{S_n{}^{n+1}\}=\{RLD\}+\{QB\} \tag{3.6.2.17}$$

where

$$[CMATRX]=\frac{[CMATRX1]}{\Delta t}+W_1[CMATRX2] \tag{3.6.2.18}$$

$$\{RLD\} = \left(\frac{[CMATRX1]}{\Delta t} - W_2[CMATRX2] \right) \{S_n{}^n\} + \{SS\} \quad \textbf{(3.6.2.19)}$$

For interior nodes i, B_i is zero, for boundary nodes i = b, B_i is calculated according to the specified boundary condition and shown as follows.

Dirichlet boundary condition

$$S_n = S_n(x_b, y_b, t) \quad \textbf{(3.6.2.20)}$$

Variable boundary condition

< Case 1 > Flow is going in from outside (**n·q** < 0).

$$\mathbf{n} \cdot \left(\mathbf{q} S_n - h\mathbf{K} \cdot \nabla S_n \right) = \mathbf{n} \cdot \mathbf{q} S_n(x_b, y_b, t) \;\; \Rightarrow \;\; B_i = -\int_B \mathbf{n} \cdot W_i \mathbf{q} S_n(x_b, y_b, t) dB \quad \textbf{(3.6.2.21)}$$

< Case 2 > Flow is going out from inside (**n·q** > 0).

$$-\mathbf{n} \cdot \left(h\mathbf{K} \cdot \nabla S_n \right) = 0 \;\; \Rightarrow \;\; B_i = -\int_B \mathbf{n} \cdot W_i \mathbf{q} S_n dB \quad \textbf{(3.6.2.22)}$$

Cauchy boundary condition

$$\mathbf{n} \cdot \left(\mathbf{q} S_n - h\mathbf{K} \cdot \nabla S_n \right) = Q_{Sn}(x_b, y_b, t) \;\; \Rightarrow \;\; B_i = -\int_B W_i Q_{Sn}(x_b, y_b, t) dB \quad \textbf{(3.6.2.23)}$$

Neumann boundary condition

$$-\mathbf{n} \cdot \left(h\mathbf{K} \cdot \nabla S_n \right) = Q_{Sn}(x_b, y_b, t) \;\; \Rightarrow \;\; B_i = -\int_B \mathbf{n} \cdot W_i \mathbf{q} S_n dB + \int_B N_i Q_{Sn}(x_b, y_b, t) dB \quad \textbf{(3.6.2.24)}$$

River/stream-overland interface boundary condition

$$\begin{aligned} &\mathbf{n} \cdot \left(\mathbf{q} S_n - h\mathbf{K} \cdot \nabla S_n \right) = \left(\mathbf{n} \cdot \mathbf{q} \right) \frac{1}{2} \left\{ \left[1 + sign\left(\mathbf{n} \cdot \mathbf{q} \right) \right] S_n + \left[1 - sign\left(\mathbf{n} \cdot \mathbf{q} \right) \right] S_n^{1D}(x_b, y_b, t) \right\} \\ &\Rightarrow \;\; B_i = -\int_B W_i \left(\mathbf{n} \cdot \mathbf{q} \right) \frac{1}{2} \left\{ \left[1 + sign\left(\mathbf{n} \cdot \mathbf{q} \right) \right] S_n + \left[1 - sign\left(\mathbf{n} \cdot \mathbf{q} \right) \right] S_n^{1D}(x_b, y_b, t) \right\} dB \end{aligned} \quad \textbf{(3.6.2.25)}$$

3.6.3 Application of the Finite Element Method to the Advective Form of the Transport Equations to Solve 2-D Suspended Sediment Transport

Conversion of the governing equation for 2-D suspended sediment transport, equation (2.6.10), to advection form is expressed as

$$h \frac{\partial S_n}{\partial t} + \mathbf{q} \cdot \nabla S_n - \nabla \cdot (h\mathbf{K} \cdot \nabla S_n) + \left(\frac{\partial h}{\partial t} + \nabla \cdot \mathbf{q} \right) S_n = M_{S_n^{as}} + M_{S_n^{rs}} + R_n - D_n \quad \textbf{(3.6.3.1)}$$

According to governing equation for 2-D water flow, equation (2.2.1), assign and calculate the right-hand side term R_{HS} and left hand side term L_{HS} as follows.

$$\begin{aligned}&\text{Assign } L_{HS} = S_S + S_R - S_E + S_I \text{ and } R_{HS} = R_n - D_n \quad \text{then continuously calculate}\\&(1): \; If \; S_S \le 0, \; L_{HS} = L_{HS} - S_S, \; ELSE \;\; R_{HS} = R_{HS} + S_S * S_n^{as}\\&(2): \; If \; S_R \le 0, \; L_{HS} = L_{HS} - S_R, \; ELSE \;\; R_{HS} = R_{HS} + S_R * S_n^{rs}\end{aligned} \quad \textbf{(3.6.3.2)}$$

Then equation (3.6.3.1) is modified as

$$h\frac{\partial S_n}{\partial t} + \mathbf{q}\cdot\nabla S_n - \nabla\cdot(h\mathbf{K}\cdot\nabla S_n) + L_{HS} * S_n = R_{HS} \quad \textbf{(3.6.3.3)}$$

Use Galerkin or Petrov-Galerkin finite-element method for the spatial descretization of transport equation. Integrate equation (3.6.3.3) in the spatial dimensions over the entire region as follows.

$$\int_R N_i\left[h\frac{\partial S_n}{\partial t} - \nabla\cdot(h\mathbf{K}\cdot\nabla S_n) + L_{HS} * S_n\right]dR + \int_R W_i\mathbf{q}\cdot\nabla S_n dR = \int_R N_i R_{HS} dR \quad \textbf{(3.6.3.4)}$$

Further, we obtain

$$\begin{aligned}&\int_R N_i h\frac{\partial S_n}{\partial t}dR + \int_R W_i\mathbf{q}\cdot\nabla S_n dR + \int_R \nabla N_i\cdot(h\mathbf{K}\cdot\nabla S_n)\,dR + \int_R N_i\,L_{HS} * S_n dR\\&\quad = \int_R N_i R_{HS} dR + \int_B \mathbf{n}\cdot N_i(h\mathbf{K}\cdot\nabla S_n)\,dB\end{aligned} \quad \textbf{(3.6.3.5)}$$

Approximate solution S_n by a linear combination of the base functions as shown by equation (3.6.3.6).

$$S_n \approx \hat{S}_n = \sum_{j=1}^{N} S_{nj}(t)N_j(R) \quad \textbf{(3.6.3.6)}$$

Substituting equation (3.6.3.6) into equation (3.6.3.5), we obtain

$$\begin{aligned}&\sum_{j=1}^{N}\left\{\left[\int_R N_i L_{HS} N_j dR + \int_R W_i\mathbf{q}\cdot\nabla N_j dR + \int_R \nabla N_i\cdot\left(h\mathbf{K}\cdot\nabla N_j\right)dR\right]S_{nj}(t)\right\}\\&+\sum_{j=1}^{N}\left[\left(\int_R N_i h N_j dR\right)\frac{dS_{nj}(t)}{dt}\right] = \int_R N_i R_{HS} dR + \int_B \mathbf{n}\cdot(N_i h\mathbf{K}\cdot\nabla S_n)dB\end{aligned} \quad \textbf{(3.6.3.7)}$$

Equation (3.6.3.7) can be written in matrix form as

$$[CMATRX1]\left\{\frac{dS_n}{dt}\right\} + ([Q1]+[Q2]+[Q3])\{S_n\} = \{SS\} + \{B\} \quad \textbf{(3.6.3.8)}$$

where the matrices [CMATRX1], [Q1], [Q2], [Q3] and load vectors {RLD}, and {B} are given by

$$CMATRX1_{ij} = \int_R N_i h N_j dR \quad \textbf{(3.6.3.9)}$$

$$Q1_{ij} = \int_R N_i L_{HS} N_j dR \quad \textbf{(3.6.3.10)}$$

$$Q2_{ij} = \int_R W_i\mathbf{q}\cdot\nabla N_j dR \quad \textbf{(3.6.3.11)}$$

$$Q3_{ij} = -\int_R \nabla N_i \cdot h\mathbf{K} \cdot \nabla N_j dR \quad \textbf{(3.6.3.12)}$$

$$SS_{ij} = \int_R N_i R_{HS} dR \quad \textbf{(3.6.3.13)}$$

$$B_i = \int_B \mathbf{n} \cdot (N_i h\mathbf{K} \cdot \nabla S_n) dB \quad \textbf{(3.6.3.14)}$$

where all the integrations are evaluated with the corresponding time weighting values.

At n+1-th time step, equation (3.6.3.8) is approximated as

$$[CMATRX1]\left\{\frac{S_n{}^{n+1} - S_n{}^{n}}{\Delta t}\right\} + [CMATRX2]\{W_1 S_n{}^{n+1} + W_2 S_n{}^{n}\} = \{SS\} + \{B\} \quad \textbf{(3.6.3.15)}$$

where

$$[CMATRX2] = [Q1] + [Q2] + [Q3] \quad \textbf{(3.6.3.16)}$$

So that

$$[CMATRX]\{S_n{}^{n+1}\} = \{RLD\} + \{QB\} \quad \textbf{(3.6.3.17)}$$

where

$$[CMATRX] = \frac{[CMATRX1]}{\Delta t} + W_1[CMATRX2] \quad \textbf{(3.6.3.18)}$$

$$\{RLD\} = \left(\frac{[CMATRX1]}{\Delta t} - W_2[CMATRX2]\right)\{S_n{}^{n}\} + \{SS\} \quad \textbf{(3.6.3.19)}$$

For interior nodes i, B_i is zero, for boundary nodes i = b, B_i is calculated according to the specified boundary condition and shown as follows.

Dirichlet boundary condition

$$S_n = S_n(x_b, y_b, t) \tag{3.6.3.20}$$

Variable boundary condition

< Case 1 > when flow is going in from outside ($\mathbf{n}\cdot\mathbf{q} < 0$)

$$\mathbf{n}\cdot\left(\mathbf{q}S_n - h\mathbf{K}\cdot\nabla S_n\right) = \mathbf{n}\cdot\mathbf{q}S_n(x_b, y_b, t) \;\Rightarrow\; B_i = \int_B \mathbf{n}\cdot N_i\mathbf{q}S_n dB - \int_B \mathbf{n}\cdot N_i\mathbf{q}S_n(x_b, y_b, t)dB \tag{3.6.3.21}$$

< Case 2 > Flow is going out from inside ($\mathbf{n}\cdot\mathbf{q} > 0$):

$$-\mathbf{n}\cdot\left(h\mathbf{K}\cdot\nabla S_n\right) = 0 \;\Rightarrow\; B_i = 0 \tag{3.6.3.22}$$

Cauchy boundary condition

$$\mathbf{n}\cdot\left(qS_n - h\mathbf{K}\cdot\nabla S_n\right) = Q_{Sn}(x_b, y_b, t) \;\Rightarrow\; B_i = \int_B N_i\mathbf{n}\cdot\mathbf{q}S_n dB - \int_B N_i Q_{Sn}(x_b, y_b, t)dB \tag{3.6.3.23}$$

Neumann boundary condition

$$-\mathbf{n}\cdot\left(h\mathbf{K}\cdot\nabla S_n\right) = Q_{Sn}(x_b, y_b, t) \;\Rightarrow\; B_i = -\int_B N_i Q_{Sn}(x_b, y_b, t)dB \tag{3.6.3.24}$$

River/stream-overland interface boundary condition

$$\begin{aligned}&\mathbf{n}\cdot\left(\mathbf{q}S_n - h\mathbf{K}\cdot\nabla S_n\right) = \left(\mathbf{n}\cdot\mathbf{q}\right)\frac{1}{2}\left\{\left[1+sign\left(\mathbf{n}\cdot\mathbf{q}\right)\right]S_n + \left[1-sign\left(\mathbf{n}\cdot\mathbf{q}\right)\right]S_n^{1D}(x_b, y_b, t)\right\}\\ &\Rightarrow\; B_i = \int_B N_i\mathbf{n}\cdot\mathbf{q}S_n dB - \int_B N_i\left(\mathbf{n}\cdot\mathbf{q}\right)\frac{1}{2}\left\{\left[1+sign\left(\mathbf{n}\cdot\mathbf{q}\right)\right]S_n + \left[1-sign\left(\mathbf{n}\cdot\mathbf{q}\right)\right]S_n^{1D}(x_b, y_b, t)\right\}dB\end{aligned} \tag{3.6.3.25}$$

3.6.4 Application of the Modified Lagrangian-Eulerian Approach to the Largrangian Form of the Transport Equations to Solve 2-D Suspended Sediment Transport

Recall governing equation for 2-D suspended sediment transport in advection form, equation (3.6.3.1), as follows

$$h\frac{\partial S_n}{\partial t} + \mathbf{q}\cdot\nabla \mathrm{S_n} - \nabla\cdot(\mathrm{h}\mathbf{K}\cdot\nabla \mathrm{S_n}) + \left(\frac{\partial h}{\partial t} + \nabla\cdot\mathbf{q}\right)\mathrm{S_n} = M_n^{as} + R_n - D_n \tag{3.6.4.1}$$

Assign and calculate R_{HS} and L_{HS} in the same way as that in section 3.6.3. Then equation (3.6.4.1) is simplified as

$$h\frac{\partial S_n}{\partial t} + \mathbf{q}\cdot\nabla \mathrm{S_n} - \nabla\cdot(\mathrm{h}\mathbf{K}\cdot\nabla \mathrm{S_n}) + L_{HS} * \mathrm{S_n} = R_{HS} \tag{3.6.4.2}$$

Equation (3.6.4.2) in the Lagrangian and Eulerian form is written as follows.

In lagrangian step,

$$h\frac{dS_n}{d\tau} = h\frac{\partial S_n}{\partial t} + \mathbf{q}\cdot\nabla S_n = 0 \quad \Rightarrow \quad \frac{\partial S_n}{\partial t} + \mathbf{v}\cdot\nabla S_n = 0 \tag{3.6.4.3}$$

where particle-tracking velocity **v** is the flow velocity.

In Eulerian step,

$$h\frac{dS_n}{d\tau} - \nabla\cdot(\mathrm{h}\mathbf{K}\cdot\nabla \mathrm{S_n}) + L_{HS} * \mathrm{S_n} = R_{HS} \tag{3.6.4.4}$$

where Δτ is the tracking time, * corresponds to the previous time step value at the location where node i is tracked through particle tracking in Lagrangian step.

Equation (3.6.4.4) written in a slightly different form is shown as

$$\frac{dS_n}{d\tau} - D + K * \mathrm{S_n} = RL \tag{3.6.4.5}$$

where

$$D = \frac{1}{h}\nabla\cdot(h\mathbf{K}\cdot\nabla S_n) \tag{3.6.4.6}$$

$$K = \frac{L_{HS}}{h} \tag{3.6.4.7}$$

$$RL = \frac{R_{HS}}{h} \tag{3.6.4.8}$$

Equation (3.6.4.5) written in matrix form is then expressed as

$$\begin{aligned}&\frac{[U]}{\Delta\tau}\left\{S_n^{\ n+1}\right\} - W_1\left\{D^{n+1}\right\} + W_1\left[K^{n+1}\right]\left\{S_n^{\ n+1}\right\} = \\ &\frac{[U]}{\Delta\tau}\left\{S_n^{\ *}\right\} + W_2\left\{D^*\right\} - W_2\left\{\left(KS_n\right)^*\right\} + W_1\left\{RL^{n+1}\right\} + W_2\left\{RL^*\right\}\end{aligned} \tag{3.6.4.9}$$

where $[K^{n+1}]$ is a diagonal matrix with K calculated at n+1-th time step as its diagonal components..

The diffusion term D expressed in term of S_n is solved by the following procedure.

Approximate D by a linear combination of the base functions as follows.

$$D \approx \hat{D} = \sum_{j=1}^{N} D_j(t) N_j(R) \tag{3.6.4.10}$$

where N is the number of nodes. According to equation (3.6.4.6), the integration of equation (3.6.4.10) can be written as

$$\int_R N_i hDdR = \int_R N_i h \sum_{j=1}^{N} D_j(t) N_j(R) dR = \int_R N_i \nabla\cdot(h\mathbf{K}\cdot\nabla S_n) dR \tag{3.6.4.11}$$

Further, we obtain

$$\sum_{j=1}^{N}\left[\left(\int_R N_i h N_j dR\right) * D_j\right] = -\int_R \nabla N_i \cdot (h\mathbf{K}\cdot\nabla S_n) dR + \int_B \mathbf{n}\cdot N_i (h\mathbf{K}\cdot\nabla S_n) dB \quad \textbf{(3.6.4.12)}$$

Approximate S_n by a linear combination of the base functions as follows.

$$S_n \approx \widehat{S}_n = \sum_{j=1}^{N} S_{nj}(t) N_j(R) \quad \textbf{(3.6.4.13)}$$

Equation (3.6.4.12) is further expressed as

$$\sum_{j=1}^{N}\left[\left(\int_R N_i h N_j dR\right) * D_j\right] = -\sum_{j=1}^{N}\left[\left(\int_R \nabla N_i \cdot (h\mathbf{K}\cdot\nabla N_j) dR\right) * (S_n)_j\right] + \int_B \mathbf{n}\cdot N_i (h\mathbf{K}\cdot\nabla S_n) dB \quad \textbf{(3.6.4.14)}$$

Assign matrices [QA] and [QD] and load vector {QB} as following.

$$QA_{ij} = \int_R N_i h N_j dR \quad \textbf{(3.6.4.15)}$$

$$QD_{ij} = \int_R \nabla N_i \cdot (h\mathbf{K}\cdot\nabla N_j) dR \quad \textbf{(3.6.4.16)}$$

$$QB_i = \int_B \mathbf{n}\cdot N_i (h\mathbf{K}\cdot\nabla S_n) dB \quad \textbf{(3.6.4.17)}$$

Equation (3.6.4.14) is expressed as

$$[QA]\{D\} = -[QD]\{S_n\} + \{QB\} \quad \textbf{(3.6.4.18)}$$

Lump matrix [QA] into diagonal matrix and update

$$QD_{ij} = QD_{ij} / QA_{ii} \quad \textbf{(3.6.4.19)}$$

$$B_i = QB_i / QA_{ii} \quad \textbf{(3.6.4.20)}$$

Then

$$\{D\} = -[QD]\{S_n\} + \{B\} \quad \textbf{(3.6.4.21)}$$

According to equation (3.6.4.21), Equation (3.6.4.9) can be modified as following

$$[CMATRX]\{S_n^{\,n+1}\} = \{RLD\} \quad \textbf{(3.6.4.22)}$$

where

$$[CMATRX] = \frac{[U]}{\Delta\tau} + W_1[QD^{n+1}] + W_1\left[K^{n+1}\right] \quad \textbf{(3.6.4.23)}$$

$$\{RLD\} = \frac{[U]}{\Delta\tau}\{S_n^{\,*}\} + W_2\{D^*\} - W_2\{(KS_n)^*\} + W_1\{RL^{n+1}\} + W_2\{RL^*\} + W_1\{B^{n+1}\} \quad \textbf{(3.6.4.24)}$$

For interior nodes, the boundary term {B} is zero. For boundary node i = b, {B} should be calculated as follows.

Dirichlet boundary condition

$$S_n = S_n(x_b, y_b, t) \;\;\Rightarrow\;\; B_i = \int_B \mathbf{n} \cdot N_i (h\mathbf{K} \cdot \nabla S_n) dB \Big/ QA_{ii} \tag{3.6.4.25}$$

Variable boundary condition

< Case 1 > when flow is going in from outside ($\mathbf{n} \cdot \mathbf{q} < 0$)

$$\begin{aligned} &\mathbf{n} \cdot \left(\mathbf{q} S_n - h\mathbf{K} \cdot \nabla S_n\right) = \mathbf{n} \cdot \mathbf{q} S_n(x_b, y_b, t) \\ \Rightarrow\;\; &B_i = \int_B \mathbf{n} \cdot N_i \mathbf{q} S_n dB \Big/ QA_{ii} - \int_B \mathbf{n} \cdot N_i \mathbf{q} S_n(x_b, y_b, t) dB \Big/ QA_{ii} \end{aligned} \tag{3.6.4.26}$$

< Case 2 > Flow is going out from inside ($\mathbf{n} \cdot \mathbf{q} > 0$):

$$-\mathbf{n} \cdot \left(h\mathbf{K} \cdot \nabla S_n\right) = 0 \;\;\Rightarrow\;\; B_i = 0 \tag{3.6.4.27}$$

Cauchy boundary condition

$$\begin{aligned} &\mathbf{n} \cdot \left(\mathbf{q} S_n - h\mathbf{K} \cdot \nabla S_n\right) = Q_{Sn}(x_b, y_b, t) \\ \Rightarrow\;\; &B_i = \int_B N_i \mathbf{n} \cdot \mathbf{q} S_n dB \Big/ QA_{ii} - \int_B N_i Q_{Sn}(x_b, y_b, t) dB \Big/ QA_{ii} \end{aligned} \tag{3.6.4.28}$$

Neumann boundary condition

$$-\mathbf{n} \cdot \left(h\mathbf{K} \cdot \nabla S_n\right) = Q_{Sn}(x_b, y_b, t) \;\;\Rightarrow\;\; B_i = -\int_B N_i Q_{Sn}(x_b, y_b, t) dB \Big/ QA_{ii} \tag{3.6.4.29}$$

River/stream-overland interface boundary condition

$$\begin{aligned} &\mathbf{n} \cdot \left(\mathbf{q} S_n - h\mathbf{K} \cdot \nabla S_n\right) = \left(\mathbf{n} \cdot \mathbf{q}\right) \frac{1}{2} \left\{\left[1 + sign\left(\mathbf{n} \cdot \mathbf{q}\right)\right] S_n + \left[1 - sign\left(\mathbf{n} \cdot \mathbf{q}\right)\right] S_n^{1D}(x_b, y_b, t)\right\} \\ \Rightarrow\;\; &B_i = \int_B N_i \mathbf{n} \cdot \mathbf{q} S_n dB \Big/ QA_{ii} - \int_B N_i \left(\mathbf{n} \cdot \mathbf{q}\right) \frac{1}{2} \left\{\left[1 + sign\left(\mathbf{n} \cdot \mathbf{q}\right)\right] S_n + \left[1 - sign\left(\mathbf{n} \cdot \mathbf{q}\right)\right] S_n^{1D}(x_b, y_b, t)\right\} dB \Big/ QA_{ii} \end{aligned} \tag{3.6.4.30}$$

At upstream flux boundary nodes, equation (3.6.4.9) cannot be applied because $\Delta\tau$ equals zero. Thus, we propose a modified LE approach in which the matrix equation for upstream boundary nodes is obtained by explicitly applying the finite element method to the boundary conditions. Applying FEM at the upstream variable boundary side, we get

$$\int_B N_i \mathbf{n} \cdot (\mathbf{q} S_n - h\mathbf{K} \cdot \nabla S_n) dB = \int_B N_i \mathbf{n} \cdot \mathbf{q} S_n(x_b, y_b, t) dB \tag{3.6.4.31}$$

So that the following matrix equation can be assembled at the upstream variable boundary node

$$[QF]\{S_n\} = [QB]\{B\} \tag{3.6.4.32}$$

in which

$$QF_{ij} = \int_B (N_i \mathbf{n} \cdot \mathbf{q} N_j - N_i \mathbf{n} \cdot h\mathbf{K} \cdot \nabla N_j) dB \qquad \textbf{(3.6.4.33)}$$

$$QB_{ij} = \int_B N_i \mathbf{n} \cdot \mathbf{q} N_j dB \qquad \textbf{(3.6.4.34)}$$

$$B_i = S_n(x_b, y_b, t) \qquad \textbf{(3.6.4.35)}$$

Similarly, equation (3.6.2.32) can be applied to Cauchy boundary with [QB] and {B} defined differently as

$$QB_{ij} = \int_B N_i N_j dB \qquad \textbf{(3.6.4.36)}$$

$$B_i = Q_{S_n}(x_b, y_b, t) \qquad \textbf{(3.6.4.37)}$$

At upstream river/stream-overland interface boundary, [QB] is calculated by equation (3.6.2.34), and {B} is defined as

$$B_i = S_n^{1D}(x_b, y_b, t) \qquad \textbf{(3.6.4.38)}$$

3.6.5 Application of the Lagrangian-Eulerian Approach for All Interior Nodes and Downstream Boundary Nodes with the Finite Element Method Applied to the Conservative Form of the Transport Equations for the Upstream Flux Boundaries to Solve 2-D Suspended Sediment Transport

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.6.4, and the matrix equation for upstream boundary nodes is obtained through the same procedure as that in section 3.6.2.

3.6.6 Application of the Lagrangian-Eulerian Approach for All Interior Nodes and Downstream Boundary Nodes with the Finite Element Method Applied to the Advective Form of the Transport Equations for the Upstream Flux Boundaries to Solve 2-D Suspended Sediment Transport

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.6.4, and the matrix equation for upstream boundary nodes is obtained through the same procedure as that in section 3.6.3.

3.6.7 Application of the Finite Element Method to the Conservative Form of the Transport Equations to Solve 2-D Kinetic Variable Transport

3.6.7.1 Fully-implicit scheme

Recall the governing equation for 2-D kinetic variable transport, equation (2.6.46), as follows

$$h\frac{\partial E_n}{\partial t}+\frac{\partial h}{\partial t}E_n+\nabla\cdot(\mathbf{q}E_n{}^m)-\nabla\cdot(h\mathbf{K}\cdot\nabla E_n{}^m)=M_{E_n{}^{as}}+M_{E_n{}^{rs}}+M_{E_n{}^{is}}+hR_{E_n},\ n\in[1,\ M-N_E] \quad \textbf{(3.6.7.1.1)}$$

At n+1-th time step, equation (3.6.7.1.1) is approximated by

$$h\frac{(E_n)^{n+1}-(E_n)^n}{\Delta t}+\frac{\partial h}{\partial t}E_n+\nabla\cdot(qE_n{}^m)-\nabla\cdot(hK\cdot\nabla E_n{}^m)=M_{E_n{}^{as}}+M_{E_n{}^{rs}}+M_{E_n{}^{is}}+hR_{E_n} \quad \textbf{(3.6.7.1.2)}$$

where the superscripts n and n+1 represent the time step number. Terms without superscript should be the corresponding average values calculated with time weighting factors W_1 and W_2.

According to Fully-implicit scheme, equation (3.6.7.1.2) can be separated into two equations as follows

$$h\frac{(E_n)^{n+1/2}-(E_n)^n}{\Delta t}+\frac{\partial h}{\partial t}E_n+\nabla\cdot(\mathbf{q}E_n{}^m)-\nabla\cdot(h\mathbf{K}\cdot\nabla E_n{}^m)=M_{E_n{}^{as}}+M_{E_n{}^{rs}}+M_{E_n{}^{is}}+hR_{E_n} \quad \textbf{(3.6.7.1.3)}$$

$$\frac{(E_n)^{n+1}-(E_n)^{n+1/2}}{\Delta t}=0 \quad \textbf{(3.6.7.1.4)}$$

First, we express $E_n{}^m$ in terms of $(E_n{}^m/E_n)\cdot E_n$ to make E_n's as primary dependent variables, so that $E_n{}^{n+1/2}$ can be solved through equation (3.6.7.1.3). Second, we solve equation (3.6.7.1.4) together with algebraic equations for equilibrium reactions using BIOGEOCHEM to obtain all individual species concentrations. Iteration between these two steps is needed because the new reaction terms $RA_n{}^{n+1}$ and the equation coefficients in equation (3.6.7.1.3) need to be updated by the calculation results of (3.6.7.1.4). To improve the standard SIA method, the nonlinear reaction terms are approximated by the Newton-Raphson linearization.

To solve equation (3.6.7.1.3), assign

$$R_{HS}=0 \quad and \quad L_{HS}=0 \quad \textbf{(3.6.7.1.5)}$$

Then the right hand side R_{HS} and left hand side L_{HS} should be continuously calculated as following.

$$M_{E_n}{}^{rs}=\begin{cases}S_R*E_{n^{rs}},\ \ if\ S_R>0\ \Rightarrow\ R_{HSn}=R_{HSn}+M_{E_n}{}^{rs}\\ S_R*E_n{}^m,\ \ if\ S_R\le 0\ \Rightarrow\ L_{HSn}=L_{HSn}-S_R\end{cases} \quad \textbf{(3.6.7.1.6)}$$

$$M_{E_n}{}^{as}=\begin{cases}S_S*E_{n^{as}},\ \ if\ S_S>0\ \Rightarrow\ R_{HSn}=R_{HSn}+M_{E_n}{}^{as},\\ S_S*E_n{}^m,\ \ if\ S_S\le 0\ \Rightarrow\ L_{HSn}=L_{HSn}-S_S\end{cases} \quad \textbf{(3.6.7.1.7)}$$

$$M_{E_n}{}^{is}=\begin{cases}S_I*E_n{}^m{}_{is},\ \ if\ S_I>0\ \Rightarrow\ R_{HSn}=R_{HSn}+M_{E_n}{}^{is}\\ S_I*E_n{}^m,\ \ if\ S_I\le 0\ \Rightarrow\ L_{HSn}=L_{HSn}-S_I\end{cases} \quad \textbf{(3.6.7.1.8)}$$

Equation (3.6.7.1.3) is then simplified as:

$$h\frac{(E_n)^{n+1/2}-(E_n)^n}{\Delta t}+\frac{\partial h}{\partial t}E_n+\nabla\cdot(\mathbf{q}E_n{}^m)-\nabla\cdot(h\mathbf{K}\cdot\nabla E_n{}^m)+L_{HS}E_n{}^m=R_{HS}+hR_{E_n} \qquad \textbf{(3.6.7.1.9)}$$

Express $E_n{}^m$ in terms of $(E_n{}^m/E_n)\,E_n{}^m$ to make E_n's as primary dependent variables,

$$\begin{aligned}&h\frac{(E_n)^{n+1/2}-(E_n)^n}{\Delta t}+\nabla\cdot\left(\mathbf{q}\frac{E_n{}^m}{E_n}E_n\right)-\nabla\cdot\left(h\mathbf{K}\cdot\frac{E_n{}^m}{E_n}\nabla E_n\right)\\&-\nabla\cdot\left[h\mathbf{K}\cdot\left(\nabla\frac{E_n{}^m}{E_n}\right)E_n\right]+\left(L_{HS}\frac{E_n{}^m}{E_n}+\frac{\partial h}{\partial t}\right)E_n=R_{HS}+hR_{E_n}\end{aligned} \qquad \textbf{(3.6.7.1.10)}$$

Use Galerkin or Petrov-Galerkin finite-element method for the spatial descretization of transport equation. Integrate equation (3.6.7.1.10) in the spatial dimensions over the entire region as follows.

$$\begin{aligned}&\int_R N_i\left[h\frac{\partial E_n}{\partial t}-\nabla\cdot\left(h\mathbf{K}\cdot\frac{E_n{}^m}{E_n}\nabla E_n\right)\right]dR+\int_R W_i\left\{\nabla\cdot\left(\mathbf{q}\frac{E_n{}^m}{E_n}E_n\right)-\nabla\cdot\left[h\mathbf{K}\cdot\left(\nabla\frac{E_n{}^m}{E_n}\right)E_n\right]\right\}dR\\&+\int_R N_i\left(L_{HS}\frac{E_n{}^m}{E_n}+\frac{\partial h}{\partial t}\right)E_n dR=\int_R N_i(R_{HS}+hR_{E_n})dR\end{aligned} \qquad \textbf{(3.6.7.1.11)}$$

Further, we obtain

$$\begin{aligned}&\int_R N_i h\frac{\partial E_n}{\partial t}dR-\int_R\nabla W_i\cdot\mathbf{q}\frac{E_n{}^m}{E_n}E_n dR+\int_R\nabla N_i\cdot\left(h\mathbf{K}\cdot\frac{E_n{}^m}{E_n}\nabla E_n\right)dR+\int_R\nabla W_i\cdot\left[h\mathbf{K}\cdot\left(\nabla\frac{E_n{}^m}{E_n}\right)E_n\right]dR\\&+\int_R N_i\left(L_{HS}\frac{E_n{}^m}{E_n}+\frac{\partial h}{\partial t}\right)E_n dR=\int_R N_i(R_{HS}+hR_{E_n})dR\\&-\int_B\mathbf{n}\cdot W_i\mathbf{q}\frac{E_n{}^m}{E_n}E_n dB+\int_B\mathbf{n}\cdot\left(N_i h\mathbf{K}\cdot\frac{E_n{}^m}{E_n}\nabla E_n\right)dB+\int_B\mathbf{n}\cdot\left[W_i h\mathbf{K}\cdot\left(\nabla\frac{E_n{}^m}{E_n}\right)E_n\right]dB\end{aligned} \qquad \textbf{(3.6.7.1.12)}$$

Approximate solution E_n by a linear combination of the base functions as follows

$$E_n\approx\hat{E}_n=\sum_{j=1}^{N}E_{nj}(t)N_j(R) \qquad \textbf{(3.6.7.1.13)}$$

Substituting Equation (3.6.7.1.13) into Equation (3.6.7.1.12), we obtain

$$\begin{aligned}&\sum_{j=1}^{N}\left\{\begin{bmatrix}-\int_R\nabla W_i\cdot\mathbf{q}\frac{E_n{}^m}{E_n}N_j dR+\int_R\nabla W_i\cdot\left[h\mathbf{K}\cdot\left(\nabla\frac{E_n{}^m}{E_n}\right)N_j\right]dR\\+\int_R\nabla N_i\cdot\left(h\mathbf{K}\cdot\frac{E_n{}^m}{E_n}\nabla N_j\right)dR+\int_R N_i\left(L_{HS}\frac{E_n{}^m}{E_n}+\frac{\partial h}{\partial t}\right)N_j dR\end{bmatrix}E_{nj}(t)\right\}\\&+\sum_{j=1}^{N}\left[\left(\int_R N_i h N_j dR\right)\frac{\partial E_{nj}(t)}{\partial t}\right]=\int_R N_i\left(R_{HS}+hR_{E_n}\right)dR\\&-\int_B\mathbf{n}\cdot W_i\mathbf{q}\frac{E_n{}^m}{E_n}E_n dB+\int_B\mathbf{n}\cdot\left(N_i h\mathbf{K}\cdot\frac{E_n{}^m}{E_n}\nabla E_n\right)dB+\int_B\mathbf{n}\cdot\left[W_i h\mathbf{K}\cdot\left(\nabla\frac{E_n{}^m}{E_n}\right)E_n\right]dB\end{aligned} \qquad \textbf{(3.6.7.1.14)}$$

Equation (3.6.7.1.14) can be written in matrix form as

$$[CMATRX1]\left\{\frac{\partial E_n}{\partial t}\right\}+\left([Q1]+[Q2]+[Q3]+[Q4]\right)\left\{E_n\right\}=\left\{SS\right\}+\left\{B\right\} \qquad \textbf{(3.6.7.1.15)}$$

The matrices [CMATRX1], [Q1], [Q2], [Q3], [Q4], and load vectors {SS}, {B} are given by

$$CMATRX1_{ij} = \int_R N_i h N_j dR \quad \textbf{(3.6.7.1.16)}$$

$$Q1_{ij} = -\int_R \nabla W_i \cdot \mathbf{q} \frac{E_n^{\ m}}{E_n} N_j dR \quad \textbf{(3.6.7.1.17)}$$

$$Q2_{ij} = \int_R \nabla W_i \cdot \left[h\mathbf{K} \cdot \left(\nabla \frac{E_n^{\ m}}{E_n} \right) N_j \right] dR \quad \textbf{(3.6.7.1.18)}$$

$$Q3_{ij} = \int_R \nabla N_i \cdot \left(h\mathbf{K} \cdot \frac{E_n^{\ m}}{E_n} \nabla N_j \right) dR \quad \textbf{(3.6.7.1.19)}$$

$$Q4_{ij} = \int_R N_i \left(L_{HS} \frac{E_n^{\ m}}{E_n} + \frac{\partial h}{\partial t} \right) N_j dR \quad \textbf{(3.6.7.1.20)}$$

$$SS_i = \int_R N_i \left(R_{HS} + hR_{E_n} \right) dR \quad \textbf{(3.6.7.1.21)}$$

$$B_i = -\int_B \mathbf{n} \cdot W_i \mathbf{q} \frac{E_n^{\ m}}{E_n} E_n dB + \int_B \mathbf{n} \cdot \left(N_i h\mathbf{K} \cdot \frac{E_n^{\ m}}{E_n} \nabla E_n \right) dB + \int_B \mathbf{n} \cdot \left[W_i h\mathbf{K} \cdot \left(\nabla \frac{E_n^{\ m}}{E_n} \right) E_n \right] dB \quad \textbf{(3.6.7.1.22)}$$

Equation (3.6.7.1.15) is then simplified as

$$[CMATRX1]\left\{ \frac{\partial E_n}{\partial t} \right\} + [CMATRX2]\left\{ E_n \right\} = \left\{ SS \right\} + \left\{ B \right\} \quad \textbf{(3.6.7.1.23)}$$

where

$$[CMATRX2] = [Q1] + [Q2] + [Q3] + [Q4] \quad \textbf{(3.6.7.1.24)}$$

Further,

$$[CMATRX1]\frac{\left(\{E_n^{\ n+1/2}\} - \{E_n^{\ n}\} \right)}{\Delta t} + [CMATRX2]\left(W_1 \{E_n^{\ n+1/2}\} + W_2 \{E_n^{\ n}\} \right) = \left\{ SS \right\} + \left\{ B \right\} \quad \textbf{(3.6.7.1.25)}$$

So that

$$[CMATRX]\left\{ E_n^{\ n+1/2} \right\} = \{RLD\} \quad \textbf{(3.6.7.1.26)}$$

where

$$[CMATRX] = \frac{[CMATRX1]}{\Delta t} + W_1 * [CMATRX2] \quad \textbf{(3.6.7.1.27)}$$

$$\{RLD\} = \left(\frac{[CMATRX1]}{\Delta t} - W_2 * [CMATRX2] \right)\left\{ E_n^{\ n} \right\} + \left\{ SS \right\} + \left\{ B \right\} \quad \textbf{(3.6.7.1.28)}$$

For interior nodes i, B_i is zero, for boundary nodes i = b, B_i is calculated according to the specified boundary condition and shown as follows.

$$B_i = -\int_B \mathbf{n} \cdot W_i \mathbf{q} E_n^{\ m} dB + \int_B \mathbf{n} \cdot \left(N_i h\mathbf{K} \cdot \nabla E_n^{\ m} \right) dB \quad \textbf{(3.6.7.1.29)}$$

<u>*Dirichlet boundary condition*</u>

$$E_n^{\ m} = E_n^{\ m}(x_b, y_b, t) \tag{3.6.7.1.30}$$

Variable boundary condition

< Case 1 > when flow is going in from outside (**n·q** <0)

$$\mathbf{n}\cdot\left(\mathbf{q}E_n^{\ m} - h\mathbf{K}\cdot\nabla E_n^{\ m}\right) = \mathbf{n}\cdot\mathbf{q}E_n^{\ m}(x_b, y_b, t) \quad \Rightarrow \quad B_i = -\int_B \mathbf{n}\cdot W_i\mathbf{q}E_n^{\ m}(x_b, y_b, t)dB \tag{3.6.7.1.31}$$

< Case 2 > Flow is going out from inside (**n·q** > 0):

$$-\mathbf{n}\cdot\left(h\mathbf{K}\cdot\nabla E_n^{\ m}\right) = 0 \quad \Rightarrow \quad B_i = -\int_B \mathbf{n}\cdot W_i\mathbf{q}E_n^{\ m}dB \tag{3.6.7.1.32}$$

Cauchy boundary condition

$$\mathrm{n}\cdot\left(\mathbf{q}E_n^{\ m} - h\mathbf{K}\cdot\nabla E_n^{\ m}\right) = Q_{En}^{\ \ m}(x_b, y_b, t) \quad \Rightarrow \quad B_i = -\int_B W_i Q_{En}^{\ \ m}(x_b, y_b, t)dB \tag{3.6.7.1.33}$$

Neumann boundary condition

$$-\mathbf{n}\cdot\left(h\mathbf{K}\cdot\nabla E_n^{\ m}\right) = Q_{En}^{\ \ m}(x_b, y_b, t) \Rightarrow \quad B_i \quad = -\int_B \mathbf{n}\cdot W_i\mathbf{q}E_n dB - \int_B N_i Q_{En}^{\ \ m}(x_b, y_b, t)dB \tag{3.6.7.1.34}$$

River/stream-overland interface boundary condition

$$\begin{aligned}&\mathbf{n}\cdot\left(\mathbf{q}E_n^{\ m} - h\mathbf{K}\cdot\nabla E_n^{\ m}\right) = \left(\mathbf{n}\cdot\mathbf{q}\right)\frac{1}{2}\left\{\left[1+sign\left(\mathbf{n}\cdot\mathbf{q}\right)\right]E_n^{\ m} + \left[1-sign\left(\mathbf{n}\cdot\mathbf{q}\right)\right]\left(E_n^{1D}\right)^m(x_b, y_b, t)\right\}\\ &\Rightarrow \quad B_i = -\int_B W_i\left(\mathbf{n}\cdot\mathbf{q}\right)\frac{1}{2}\left\{\left[1+sign\left(\mathbf{n}\cdot\mathbf{q}\right)\right]E_n^{\ m} + \left[1-sign\left(\mathbf{n}\cdot\mathbf{q}\right)\right]\left(E_n^{1D}\right)^m(x_b, y_b, t)\right\}dB\end{aligned} \tag{3.6.7.1.35}$$

Note: In the equation (3.6.7.1.18), assign

3.6.7.2 Mixed Predictor-corrector/Operator-splitting scheme

Recall the governing equation for 2-D kinetic variable transport at n+1-th time step, equation (3.6.7.1.2), as follows

$$h\frac{(E_n)^{n+1} - (E_n)^n}{\Delta t} + \frac{\partial h}{\partial t}E_n + \nabla\cdot(\mathbf{q}E_n^{\ m}) - \nabla\cdot(h\mathbf{K}\cdot\nabla E_n^{\ m}) = M_{E_n^{\ as}} + M_{E_n^{\ rs}} + M_{E_n^{\ is}} + hR_{E_n} \tag{3.6.7.2.1}$$

According to mixed Predictor-corrector/Operator-splitting scheme, equation (3.6.7.2.1) can be separated into two equations as follows

$$\begin{aligned}&h\frac{(E_n^{\ m})^{n+1/2} - (E_n^{\ m})^n}{\Delta t} + \frac{\partial h}{\partial t}E_n^m + \nabla\cdot(\mathbf{q}E_n^{\ m}) - \nabla\cdot(h\mathbf{K}\cdot\nabla E_n^{\ m}) =\\ &M_{E_n^{\ as}} + M_{E_n^{\ rs}} + M_{E_n^{\ is}} + h\left(R_{E_n}\right)^n - \frac{\partial h}{\partial t}(E_n^{im})^n\end{aligned} \tag{3.6.7.2.2}$$

$$\frac{E_n{}^{n+1} - [(E_n{}^m)^{n+1/2} + (E_n{}^{im})^n]}{\Delta t} = hR_{E_n}{}^{n+1} - h(R_{E_n})^n - \frac{\partial \ell n(h)}{\partial t}(E_n^{im})^{n+1} + \frac{\partial \ell n(h)}{\partial t}(E_n^{im})^n \quad \textbf{(3.6.7.2.3)}$$

First, solve equation (3.6.7.2.2) and get $(E_n{}^m)^{n+1/2}$. Second, solve equation (3.6.7.2.3) together with algebraic equations for equilibrium reactions using BIOGEOCHEM scheme to obtain the individual species concentration.

Assign and calculate the right hand side R_{HS} and left hand side L_{HS} the same as that in section 3.6.7.1, equation (3.6.7.2.2) is then simplified as:

$$h\frac{(E_n{}^m)^{n+1/2} - (E_n{}^m)^n}{\Delta t} + \nabla\cdot(\mathbf{q}E_n{}^m) - \nabla\cdot(h\mathbf{K}\cdot\nabla E_n{}^m) + \left(L_{HS} + \frac{\partial h}{\partial t}\right)E_n{}^m = R_{HS} + h\left(R_{E_n}\right)^n - \frac{\partial h}{\partial t}(E_n^{im})^n \quad \textbf{(3.6.7.2.4)}$$

Use Galerkin or Petrov-Galerkin finite-element method for the spatial descretization of transport equation. Integrate equation (3.6.7.2.4) in the spatial dimensions over the entire region as follows

$$\begin{aligned}&\int_R N_i\left[h^n\frac{\partial E_n{}^m}{\partial t} - \nabla\cdot\left(h\mathbf{K}\cdot\nabla E_n{}^m\right)\right]dR + \int_R W_i\nabla\cdot\left(\mathbf{q}E_n{}^m\right)dR \\ &+ \int_R N_i\left(L_{HS} + \frac{\partial h}{\partial t}\right)E_n{}^m dR = \int_R N_i\left(R_{HS} + hR_{E_n}{}^n - \frac{\partial h}{\partial t}(E_n^{im})^n\right)dR\end{aligned} \quad \textbf{(3.6.7.2.5)}$$

Further, we obtain

$$\begin{aligned}&\int_R N_i h\frac{\partial E_n{}^m}{\partial t}dR - \int_R \nabla W_i\cdot\mathbf{q}E_n{}^m dR + \int_R \nabla N_i\cdot\left(h\mathbf{K}\cdot\nabla E_n{}^m\right)dR + \int_R N_i\left(L_{HS} + \frac{\partial h}{\partial t}\right)E_n{}^m dR \\ &= \int_R N_i\left(R_{HS} + hR_{E_n}{}^n - \frac{\partial h}{\partial t}(E_n^{im})^n\right)dR - \int_B \mathbf{n}\cdot W_i\mathbf{q}E_n{}^m dB + \int_B \mathbf{n}\cdot\left(N_i h\mathbf{K}\cdot\nabla E_n{}^m\right)dB\end{aligned} \quad \textbf{(3.6.7.2.6)}$$

Approximate solution $E_n{}^m$ by a linear combination of the base functions as follows

$$E_n{}^m \approx \hat{E}_n{}^m = \sum_{j=1}^N E_{nj}{}^m(t)N_j(R) \quad \textbf{(3.6.7.2.7)}$$

Substituting Equation (3.6.7.2.7) into Equation (3.6.7.2.6), we obtain

$$\begin{aligned}&\sum_{j=1}^N\left\{\left[-\int_R \nabla W_i\cdot\mathbf{q}N_j dR + \int_R \nabla N_i\cdot\left(h\mathbf{K}\cdot\nabla N_j\right)dR + \int_R N_i\left(L_{HS} + \frac{\partial h}{\partial t}\right)N_j dR\right]E_{nj}{}^m(t)\right\} \\ &+ \sum_{j=1}^N\left[\left(\int_R N_i h^n N_j dR\right)\frac{\partial E_{nj}{}^m(t)}{\partial t}\right] = \int_R N_i\left(R_{HS} + hR_{E_n}{}^n - \frac{\partial h}{\partial t}(E_n^{im})^n\right)dR \\ &- \int_B \mathbf{n}\cdot W_i\mathbf{q}E_n{}^m dB + \int_B \mathbf{n}\cdot\left(N_i h\mathbf{K}\cdot\nabla E_n{}^m\right)dB\end{aligned} \quad \textbf{(3.6.7.2.8)}$$

Equation (3.6.7.2.8) can be written in matrix form as

$$[CMATRX1]\left\{\frac{\partial E_n{}^m}{\partial t}\right\} + \left([Q1] + [Q3] + [Q4]\right)\left\{E_n{}^m\right\} = \{SS\} + \{B\} \quad \textbf{(3.6.7.2.9)}$$

The matrices [CMATRX1], [Q1], [Q3], [Q4], and load vectors {SS}, {B} are given by

$$CMATRX1_{ij} = \int_R N_i h N_j \, dR \tag{3.6.7.2.10}$$

$$Q1_{ij} = -\int_R \nabla W_i \cdot \mathbf{q} N_j \, dR \tag{3.6.7.2.11}$$

$$Q3_{ij} = \int_R \nabla N_i \cdot \left(h\mathbf{K} \cdot \nabla N_j \right) dR \tag{3.6.7.2.12}$$

$$Q4_{ij} = \int_R N_i \left(L_{HS} + \frac{\partial h}{\partial t} \right) N_j \, dR \tag{3.6.7.2.13}$$

$$SS_i = \int_R N_i \left(R_{HS} + hR_{E_n}{}^n - \frac{\partial h}{\partial t} (E_n^{im})^n \right) dR \tag{3.6.7.2.14}$$

$$B_i = -\int_B \mathbf{n} \cdot W_i \mathbf{q} E_n{}^m dB + \int_B \mathbf{n} \cdot \left(N_i h\mathbf{K} \cdot \nabla E_n{}^m \right) dB \tag{3.6.7.2.15}$$

Equation (3.6.7.2.9) is then simplified as

$$[CMATRX1] \left\{ \frac{\partial E_n}{\partial t} \right\} + [CMATRX2] \{ E_n \} = \{ SS \} + \{ B \} \tag{3.6.7.2.16}$$

where

$$[CMATRX2] = [Q1] + [Q3] + [Q4] \tag{3.6.7.2.17}$$

Further,

$$\begin{aligned} &[CMATRX1] \frac{\left[\{ (E_n{}^m)^{n+1/2} \} - \{ (E_n{}^m)^n \} \right]}{\Delta t} \\ &+ [CMATRX2] \left[W_1 \{ (E_n{}^m)^{n+1/2} \} + W_2 \{ (E_n{}^m)^n \} \right] = \{ SS \} + \{ B \} \end{aligned} \tag{3.6.7.2.18}$$

So that

$$[CMATRX] \left\{ (E_n{}^m)^{n+1/2} \right\} = \{ RLD \} \tag{3.6.7.2.19}$$

where

$$[CMATRX] = \frac{[CMATRX1]}{\Delta t} + W_1 * [CMATRX2] \tag{3.6.7.2.20}$$

$$\{ RLD \} = \left(\frac{[CMATRX1]}{\Delta t} - W_2 * [CMATRX2] \right) \left\{ (E_n{}^m)^n \right\} + \{ SS \} + \{ B \} \tag{3.6.7.2.21}$$

For interior nodes i, B_i is zero, for boundary nodes i = b, B_i is the same as that in section 3.6.7.1.

3.6.7.3 Operator-splitting scheme

Recall the governing equation for 2-D kinetic variable transport at n+1-th time step, equation (3.6.7.1.2), as follows

$$h \frac{(E_n)^{n+1} - (E_n)^n}{\Delta t} + \frac{\partial h}{\partial t} E_n + \nabla \cdot (\mathbf{q} E_n{}^m) - \nabla \cdot (h\mathbf{K} \cdot \nabla E_n{}^m) = M_{E_n{}^{as}} + M_{E_n{}^{rs}} + M_{E_n{}^{is}} + hR_{E_n} \tag{3.6.7.3.1}$$

According to Operator-splitting scheme, equation (3.6.7.3.1) can be separated into two equations as follows

$$h\frac{(E_n{}^m)^{n+1/2}-(E_n{}^m)^n}{\Delta t}+\frac{\partial h}{\partial t}E_n^m+\nabla\cdot(\mathbf{q}E_n{}^m)-\nabla\cdot(h\mathbf{K}\cdot\nabla E_n{}^m)=M_{E_n^{as}}+M_{E_n^{rs}}+M_{E_n^{is}} \quad \textbf{(3.6.7.3.2)}$$

$$\frac{(E_n)^{n+1}-[(E_n{}^m)^{n+1/2}+(E_n{}^{im})^n]}{\Delta t}=hR_{E_n}{}^{n+1}-\frac{\partial \ell nh}{\partial t}(E_n^{im})^{n+1} \quad \textbf{(3.6.7.3.3)}$$

First, solve equation (3.6.7.3.2) and get $(E_n{}^m)^{n+1/2}$. Second, solve equation (3.6.7.3.3) together with algebraic equations for equilibrium reactions using BIOGEOCHEM scheme to obtain the individual species concentration.

Equation (3.6.7.3.2) can be solved through the same procedure as that in section 3.6.7.2, except for the load vectors {SS}, which is calculated by the following equation.

$$SS_i=\sum_{e=1}^{M_e}\int_{R_e}N_i{}^e R_{HS}dR \quad \textbf{(3.6.7.3.4)}$$

3.6.8 Application of the Finite Element Method to the Advective Form of the Transport Equations to Solve 2-D Kinetic Variable Transport

3.6.8.1 Fully-implicit scheme

Conversion of the equation for 2-D kinetic variable transport Fully-implicit scheme transport step, equation (3.6.7.1.3), to advection form is expressed as

$$\begin{aligned}&h\frac{(E_n)^{n+1/2}-(E_n)^n}{\Delta t}+\frac{\partial h}{\partial t}E_n+\mathbf{q}\cdot\nabla E_n{}^m-\nabla\cdot(h\mathbf{K}\cdot\nabla E_n{}^m)+(\nabla\cdot\mathbf{q})E_n{}^m=\\&M_{E_n^{as}}+M_{E_n^{rs}}+M_{E_n^{is}}+hR_{E_n}\end{aligned} \quad \textbf{(3.6.8.1.1)}$$

where $\partial h/\partial t+\nabla\cdot\mathbf{q}=S_S+S_R+S_I$ according to governing equation for 2-D flow.

To solve equation (3.6.8.1.1), assign

$$R_{HS}=0 \quad and \quad L_{HS}=S_S+S_R+S_I-\partial h/\partial t \quad \textbf{(3.6.8.1.2)}$$

Then the right hand side R_{HS} and left hand side L_{HS} should be continuously calculated the same as that in section 3.6.7.1. Equation (3.6.8.1.1) is then simplified as:

$$h\frac{\partial E_n}{\partial t}+\frac{\partial h}{\partial t}E_n+\mathbf{q}\cdot\nabla E_n{}^m-\nabla\cdot(h\mathbf{K}\cdot\nabla E_n{}^m)+L_{HS}E_n{}^m=R_{HS}+hR_{E_n} \quad \textbf{(3.6.8.1.3)}$$

Express $E_n{}^m$ in terms of $(E_n{}^m/E_n)\,E_n{}^m$ to make E_n's as primary dependent variables,

$$\begin{aligned}&h\frac{\partial E_n}{\partial t}+\mathbf{q}\cdot\nabla\left(\frac{E_n{}^m}{E_n}E_n\right)-\nabla\cdot\left(h\mathbf{K}\cdot\frac{E_n{}^m}{E_n}\nabla E_n\right)\\&-\nabla\cdot\left[h\mathbf{K}\cdot\left(\nabla\frac{E_n{}^m}{E_n}\right)E_n\right]+\left(L_{HS}\frac{E_n{}^m}{E_n}+\frac{\partial h}{\partial t}\right)E_n=R_HS+hR_{E_n}\end{aligned} \quad \textbf{(3.6.8.1.4)}$$

Use Galerkin or Petrov-Galerkin finite-element method for the spatial descretization of transport equation. Integrate equation (3.6.8.1.4) in the spatial dimensions over the entire region as follows.

$$\int_R N_i\left[h\frac{\partial E_n}{\partial t}-\nabla\cdot\left(h\mathbf{K}\cdot\frac{E_n^{\ m}}{E_n}\nabla E_n\right)\right]dR+\int_R W_i\left\{\mathbf{q}\cdot\nabla\left(\frac{E_n^{\ m}}{E_n}E_n\right)-\nabla\cdot\left[h\mathbf{K}\cdot\left(\nabla\frac{E_n^{\ m}}{E_n}\right)E_n\right]\right\}dR$$
$$+\int_R N_i\left(L_{HS}\frac{E_n^{\ m}}{E_n}+\frac{\partial h}{\partial t}\right)E_n dR=\int_R N_i(R_{HS}+hR_{E_n})dR \qquad \textbf{(3.6.8.1.5)}$$

Further, we obtain

$$\int_R N_i h\frac{\partial E_n}{\partial t}dR-\int_R W_i\mathbf{q}\cdot\nabla\frac{E_n^{\ m}}{E_n}E_n dR+\int_R\nabla N_i\cdot\left(h\mathbf{K}\cdot\frac{E_n^{\ m}}{E_n}\nabla E_n\right)dR$$
$$+\int_R\nabla W_i\cdot\left[h\mathbf{K}\cdot\left(\nabla\frac{E_n^{\ m}}{E_n}\right)E_n\right]dR+\int_R N_i\left(L_{HS}\frac{E_n^{\ m}}{E_n}+\frac{\partial h}{\partial t}\right)E_n dR \qquad \textbf{(3.6.8.1.6)}$$
$$=\int_R N_i(R_{HS}+hR_{E_n})dR+\int_B n\cdot\left(N_i hK\cdot\frac{E_n^{\ m}}{E_n}\nabla E_n\right)dB+\int_B n\cdot\left[W_i hK\cdot\left(\nabla\frac{E_n^{\ m}}{E_n}\right)E_n\right]dB$$

Approximate solution E_n by a linear combination of the base functions as follows

$$E_n\approx\hat{E}_n=\sum_{j=1}^{N}E_{nj}(t)N_j(R) \qquad \textbf{(3.6.8.1.7)}$$

Substituting Equation (3.6.8.1.7) into Equation (3.6.8.1.6), we obtain

$$\sum_{j=1}^{N}\left\{\left[\begin{array}{l}\int_R W_i\mathbf{q}\cdot\frac{E_n^{\ m}}{E_n}\nabla N_j dR+\int_R W_i\mathbf{q}\cdot\left(\nabla\frac{E_n^{\ m}}{E_n}\right)N_j dR+\int_R\nabla W_i\cdot\left[h\mathbf{K}\cdot\left(\nabla\frac{E_n^{\ m}}{E_n}\right)N_j\right]dR\\ +\int_R\nabla N_i\cdot\left(h\mathbf{K}\cdot\frac{E_n^{\ m}}{E_n}\nabla N_j\right)dR+\int_R N_i\left(L_{HS}\frac{E_n^{\ m}}{E_n}+\frac{\partial h}{\partial t}\right)N_j dR\end{array}\right]E_{nj}(t)\right\}$$
$$+\sum_{j=1}^{N}\left[\left(\int_R N_i hN_j dR\right)\frac{\partial E_{nj}(t)}{\partial t}\right]=\int_R N_i\left(R_{HS}+hR_{E_n}\right)dR \qquad \textbf{(3.6.8.1.8)}$$
$$+\int_B n\cdot\left(N_i h\mathbf{K}\cdot\frac{E_n^{\ m}}{E_n}\nabla E_n\right)dB+\int_B n\cdot\left[W_i h\mathbf{K}\cdot\left(\nabla\frac{E_n^{\ m}}{E_n}\right)E_n\right]dB$$

Equation (3.6.8.1.8) can be written in matrix form as

$$[CMATRX1]\left\{\frac{\partial E_n}{\partial t}\right\}+\left([Q1]+[Q2]+[Q3]+[Q4]+[Q5]\right)\left\{E_n\right\}=\left\{SS\right\}+\left\{B\right\} \qquad \textbf{(3.6.8.1.9)}$$

The matrices [CMATRX1], [Q1], [Q2], [Q3], [Q4], [Q5], and load vectors {SS}, {B} are given by

$$CMATRX1_{ij}=\int_R N_i hN_j dR \qquad \textbf{(3.6.8.1.10)}$$

$$Q1_{ij}=\int_R W_i\mathbf{q}\cdot\frac{E_n^{\ m}}{E_n}\nabla N_j dR \qquad \textbf{(3.6.8.1.11)}$$

$$Q2_{ij} = \int_R W_i \mathbf{q} \cdot \left(\nabla \frac{E_n{}^m}{E_n} \right) N_j dR \quad \textbf{(3.6.8.1.12)}$$

$$Q3_{ij} = \int_R \nabla W_i \cdot \left[h\mathbf{K} \cdot \left(\nabla \frac{E_n{}^m}{E_n} \right) N_j \right] dR \quad \textbf{(3.6.8.1.13)}$$

$$Q4_{ij} = \int_R \nabla N_i \cdot \left(h\mathbf{K} \cdot \frac{E_n{}^m}{E_n} \nabla N_j \right) dR \quad \textbf{(3.6.8.1.14)}$$

$$Q5_{ij} = \int_R N_i \left(L_{HS} \frac{E_n{}^m}{E_n} + \frac{\partial h}{\partial t} \right) N_j dR \quad \textbf{(3.6.8.1.15)}$$

$$SS_i = \int_R N_i \left(R_{HS} + hR_{E_n} \right) dR \quad \textbf{(3.6.8.1.16)}$$

$$B_i = \int_B \mathbf{n} \cdot \left(N_i h\mathbf{K} \cdot \frac{E_n{}^m}{E_n} \nabla E_n \right) dB + \int_B \mathbf{n} \cdot \left[W_i h\mathbf{K} \cdot \left(\nabla \frac{E_n{}^m}{E_n} \right) E_n \right] dB \quad \textbf{(3.6.8.1.17)}$$

Equation (3.6.8.1.9) is then simplified as

$$[CMATRX1]\left\{ \frac{\partial E_n}{\partial t} \right\} + [CMATRX2]\left\{ E_n \right\} = \left\{ SS \right\} + \left\{ B \right\} \quad \textbf{(3.6.8.1.18)}$$

where

$$[CMATRX2] = [Q1] + [Q2] + [Q3] + [Q4] + [Q5] \quad \textbf{(3.6.8.1.19)}$$

Further,

$$[CMATRX1]\frac{\left(\{E_n{}^{n+1/2}\} - \{E_n{}^n\} \right)}{\Delta t} + [CMATRX2]\left(W_1\{E_n{}^{n+1/2}\} + W_2\{E_n{}^n\} \right) = \left\{ SS \right\} + \left\{ B \right\} \quad \textbf{(3.6.8.1.20)}$$

So that

$$[CMATRX]\left\{ E_n{}^{n+1/2} \right\} = \{RLD\} \quad \textbf{(3.6.8.1.21)}$$

where

$$[CMATRX] = \frac{[CMATRX1]}{\Delta t} + W_1 * [CMATRX2] \quad \textbf{(3.6.8.1.22)}$$

$$\{\mathrm{RLD}\} = \left(\frac{[\mathrm{CMATRX1}]}{\Delta \mathrm{t}} - \mathrm{W_2} * [\mathrm{CMATRX2}] \right) \left\{ \mathrm{E_n}^{\mathrm{n}} \right\} + \left\{ \mathrm{SS} \right\} + \left\{ \mathrm{B} \right\} \quad \textbf{(3.6.8.1.23)}$$

For interior nodes i, B_i is zero, for boundary nodes i = b, B_i is calculated according to the specified boundary condition and shown as follows.

$$B_i = \int_B \mathbf{n} \cdot \left(N_i h\mathbf{K} \cdot \nabla E_n{}^m \right) dB \quad \textbf{(3.6.8.1.24)}$$

Dirichlet boundary condition

$$E_n{}^m = E_n{}^m(x_b, y_b, t) \quad \textbf{(3.6.8.1.25)}$$

Variable boundary condition

< Case 1 > when flow is going in from outside ($\mathbf{n}\cdot\mathbf{q} < 0$)

$$\mathbf{n}\cdot\left(\mathbf{q}E_n^{\ m} - h\mathbf{K}\cdot\nabla E_n^{\ m}\right) = \mathbf{n}\cdot\mathbf{q}E_n^{\ m}(x_b, y_b, t) \ \Rightarrow \ B_i = \int_B \mathbf{n}\cdot N_i\mathbf{q}E_n^{\ m}dB - \int_B \mathbf{n}\cdot N_i\mathbf{q}E_n^{\ m}(x_b, y_b, t)dB \qquad \textbf{(3.6.8.1.26)}$$

< Case 2 > Flow is going out from inside ($\mathbf{n}\cdot\mathbf{q} > 0$):

$$-\mathbf{n}\cdot\left(h\mathbf{K}\cdot\nabla E_n^{\ m}\right) = 0 \ \Rightarrow \ B_i = 0 \qquad \textbf{(3.6.8.1.27)}$$

<u>*Cauchy boundary condition*</u>

$$\mathbf{n}\cdot\left(\mathbf{q}E_n^{\ m} - h\mathbf{K}\cdot\nabla E_n^{\ m}\right) = Q_{En}^{\ \ m}(x_b, y_b, t) \ \Rightarrow \ B_i = \int_B N_i\mathbf{n}\cdot\mathbf{q}E_n^{\ m}dB - \int_B N_i Q_{En}^{\ \ m}(x_b, y_b, t)dB \qquad \textbf{(3.6.8.1.28)}$$

<u>*Neumann boundary condition*</u>

$$-\mathbf{n}\cdot\left(h\mathbf{K}\cdot\nabla E_n^{\ m}\right) = Q_{En}^{\ \ m}(x_b, y_b, t) \ \Rightarrow \ B_i = -\int_B N_i Q_{En}^{\ \ m}(x_b, y_b, t)dB \qquad \textbf{(3.6.8.1.29)}$$

<u>*River/stream-overland interface boundary condition*</u>

$$\begin{aligned} &\mathbf{n}\cdot\left(\mathbf{q}E_n^{\ m} - h\mathbf{K}\cdot\nabla E_n^{\ m}\right) = \left(\mathbf{n}\cdot\mathbf{q}\right)\frac{1}{2}\left\{\left[1+sign\left(\mathbf{n}\cdot\mathbf{q}\right)\right]E_n^{\ m} + \left[1-sign\left(\mathbf{n}\cdot\mathbf{q}\right)\right]\left(E_n^{1D}\right)^m(x_b, y_b, t)\right\} \ \Rightarrow \\ &B_i = \int_B N_i\mathbf{n}\cdot\mathbf{q}E_n^{\ m}dB - \int_B N_i\left(\mathbf{n}\cdot\mathbf{q}\right)\frac{1}{2}\left\{\left[1+sign\left(\mathbf{n}\cdot\mathbf{q}\right)\right]E_n^{\ m} + \left[1-sign\left(\mathbf{n}\cdot\mathbf{q}\right)\right]\left(E_n^{1D}\right)^m(x_b, y_b, t)\right\}dB \end{aligned} \qquad \textbf{(3.6.8.1.30)}$$

3.6.8.2 Mixed Predictor-corrector/Operator-splitting scheme

Conversion of the equation for 2-D kinetic variable transport mixed Predictor-corrector/Operator-splitting scheme transport step, equation (3.6.7.2.3), to advection form is expressed as

$$\begin{aligned} &h\frac{(E_n^{\ m})^{n+1/2} - (E_n^{\ m})^n}{\Delta t} + \frac{\partial h}{\partial t}E_n^m + \mathbf{q}\cdot\nabla E_n^{\ m} - \nabla\cdot(h\mathbf{K}\cdot\nabla E_n^{\ m}) + \left(\nabla\cdot\mathbf{q}\right)E_n^{\ m} = \\ &M_{E_n^{as}} + M_{E_n^{rs}} + M_{E_n^{is}} + hR_{E_n}^{\ \ n} - \frac{\partial h}{\partial t}(E_n^{im})^n \end{aligned} \qquad \textbf{(3.6.8.2.1)}$$

where $\partial h/\partial t + \nabla\cdot\mathbf{q} = S_S + S_R + S_I$ according to governing equation for 2-D flow.

To solve equation (3.6.8.2.1), assign the right hand side R_{HS} and left hand side L_{HS} the same as that in section 3.6.8.1. Equation (3.6.8.2.1) is then simplified as:

$$h\frac{\partial E_n^{\ m}}{\partial t} + \frac{\partial h}{\partial t}E_n^m + \mathbf{q}\cdot\nabla E_n^{\ m} - \nabla\cdot(h\mathbf{K}\cdot\nabla E_n^{\ m}) + L_{HS}E_n^{\ m} = R_{HS} + hR_{E_n}^{\ \ n} - \frac{\partial h}{\partial t}(E_n^{im})^n \qquad \textbf{(3.6.8.2.2)}$$

Use Galerkin or Petrov-Galerkin finite-element method for the spatial descretization of transport equation. Integrate equation (3.6.8.2.4) in the spatial dimensions over the entire region as follows.

$$\int_R N_i \left[h\frac{\partial E_n{}^m}{\partial t} - \nabla\cdot\left(h\mathbf{K}\cdot\nabla E_n{}^m\right)\right] dR + \int_R W_i \mathbf{q}\cdot\nabla E_n{}^m dR$$
$$+ \int_R N_i\left(L_{HS} + \frac{\partial h}{\partial t}\right) E_n{}^m dR = \int_R N_i \left(R_{HS} + hR_{E_n}{}^n - \frac{\partial h}{\partial t}(E_n^{im})^n\right) dR \quad \textbf{(3.6.8.2.3)}$$

Further, we obtain

$$\int_R N_i h \frac{\partial E_n{}^m}{\partial t} dR - \int_R W_i \mathbf{q}\cdot\nabla E_n{}^m dR + \int_R \nabla N_i\cdot\left(h\mathbf{K}\cdot\nabla E_n{}^m\right) dR + \int_R N_i\left(L_{HS} + \frac{\partial h}{\partial t}\right) E_n{}^m dR$$
$$= \int_R N_i \left(R_{HS} + hR_{E_n}{}^n - \frac{\partial h}{\partial t}(E_n^{im})^n\right) dR + \int_B n\cdot\left(N_i hK\cdot\nabla E_n{}^m\right) dB \quad \textbf{(3.6.8.2.4)}$$

Approximate solution $E_n{}^m$ by a linear combination of the base functions as follows

$$E_n{}^m \approx \hat{E}_n{}^m = \sum_{j=1}^{N} E_{nj}{}^m(t) N_j(R) \quad \textbf{(3.6.8.2.5)}$$

Substituting Equation (3.6.8.2.5) into Equation (3.6.8.2.4), we obtain

$$\sum_{j=1}^{N}\left\{\left[\int_R W_i \mathbf{q}\cdot\nabla N_j dR + \int_R \nabla N_i\cdot\left(h\mathbf{K}\cdot\nabla N_j\right) dR + \int_R N_i\left(L_{HS} + \frac{\partial h}{\partial t}\right) N_j dR\right] E_{nj}{}^m(t)\right\}$$
$$+\sum_{j=1}^{N}\left[\left(\int_R N_i h N_j dR\right)\frac{\partial E_{nj}{}^m(t)}{\partial t}\right] = \int_R N_i\left(R_{HS} + hR_{E_n}{}^n - \frac{\partial h}{\partial t}(E_n^{im})^n\right) dR + \int_B \mathbf{n}\cdot\left(N_i h\mathbf{K}\cdot\nabla E_n{}^m\right) dB \quad \textbf{(3.6.8.2.6)}$$

Equation (3.6.8.2.6) can be written in matrix form as

$$[CMATRX1]\left\{\frac{\partial E_n}{\partial t}\right\} + \left([Q1]+[Q4]+[Q5]\right)\left\{E_n\right\} = \left\{SS\right\} + \left\{B\right\} \quad \textbf{(3.6.8.2.7)}$$

The matrices [CMATRX1], [Q1], [Q4], [Q5], and load vectors {SS}, {B} are given by

$$CMATRX1_{ij} = \int_R N_i h N_j dR \quad \textbf{(3.6.8.2.8)}$$

$$Q1_{ij} = \int_R W_i \mathbf{q}\cdot\nabla N_j dR \quad \textbf{(3.6.8.2.9)}$$

$$Q4_{ij} = \int_R \nabla N_i\cdot\left(h\mathbf{K}\cdot\nabla N_j\right) dR \quad \textbf{(3.6.8.2.10)}$$

$$Q5_{ij} = \int_R N_i\left(L_{HS} + \frac{\partial h}{\partial t}\right) N_j dR \quad \textbf{(3.6.8.2.11)}$$

$$SS_i = \int_R N_i\left(R_{HS} + hR_{E_n}{}^n - \frac{\partial h}{\partial t}(E_n^{im})^n\right) dR \quad \textbf{(3.6.8.2.12)}$$

$$B_i = \int_B \mathbf{n}\cdot\left(N_i h\mathbf{K}\cdot\nabla E_n\right) dB \quad \textbf{(3.6.8.2.13)}$$

Equation (3.6.8.2.7) is then simplified as

$$[CMATRX1]\left\{\frac{\partial E_n}{\partial t}\right\}+[CMATRX2]\{E_n\}=\{SS\}+\{B\} \quad \textbf{(3.6.8.2.14)}$$

where

$$[CMATRX2]=[Q1]+[Q4]+[Q5] \quad \textbf{(3.6.8.2.15)}$$

Further,

$$[CMATRX1]\frac{\left(\{E_n{}^{n+1/2}\}-\{E_n{}^{n}\}\right)}{\Delta t}+[CMATRX2]\left(W_1\{E_n{}^{n+1/2}\}+W_2\{E_n{}^{n}\}\right)=\{SS\}+\{B\} \quad \textbf{(3.6.8.2.16)}$$

So that

$$[CMATRX]\left\{E_n{}^{n+1/2}\right\}=\{RLD\} \quad \textbf{(3.6.8.2.17)}$$

where

$$[CMATRX]=\frac{[CMATRX1]}{\Delta t}+W_1*[CMATRX2] \quad \textbf{(3.6.8.2.18)}$$

$$\{RLD\}=\left(\frac{[CMATRX1]}{\Delta t}-W_2*[CMATRX2]\right)\left\{E_n{}^{n}\right\}+\{SS\}+\{B\} \quad \textbf{(3.6.8.2.19)}$$

For interior nodes i, B_i is zero, for boundary nodes i = b, B_i is calculated according to the specified boundary condition calculated the same as that in section 3.6.8.1.

3.6.8.3 Operator-splitting scheme

Conversion of the equation for 2-D kinetic variable transport operator spitting scheme transport step, equation (3.6.7.3.3), to advection form is expressed as

$$h\frac{(E_n{}^m)^{n+1/2}-(E_n{}^m)^n}{\Delta t}+\frac{\partial h}{\partial t}E_n^m+\mathbf{q}\cdot\nabla E_n{}^m-\nabla\cdot(h\mathbf{K}\cdot\nabla E_n{}^m)+\left(\nabla\cdot\mathbf{q}\right)E_n{}^m=M_{E_n{}^{as}}+M_{E_n{}^{rs}}+M_{E_n{}^{is}} \quad \textbf{(3.6.8.3.1)}$$

where $\partial h/\partial t+\nabla\cdot q=S_S+S_R+S_I$ according to governing equation for 2-D flow.

Equation (3.6.8.3.1) can be solved through the same procedure as that in section 3.6.8.2, except for the load vectors {SS}, which is calculated by the following equation.

$$SS_i=\sum_{e=1}^{M_e}\int_{R_e}N_i^eR_{HS}dR \quad \textbf{(3.6.8.3.2)}$$

3.6.9 Application of the Modified Lagrangian-Eulerian Approach to the Largrangian Form of the Transport Equations to Solve 2-D Kinetic Variable Transport

3.6.9.1 Fully-implicit scheme

Recall the equation for 2-D kinetic variable transport Fully-implicit scheme transport step in advection form, equation (3.6.8.1.1), as follows

$$h\frac{(E_n)^{n+1/2}-(E_n)^n}{\Delta t}+\frac{\partial h}{\partial t}E_n+\mathbf{q}\cdot\nabla E_n{}^m-\nabla\cdot(h\mathbf{K}\cdot\nabla E_n{}^m)+\left(\nabla\cdot\mathbf{q}\right)E_n{}^m=M_{E_n{}^{as}}+M_{E_n{}^{rs}}+M_{E_n{}^{is}}+hR_{E_n} \quad \textbf{(3.6.9.1.1)}$$

Express E_n^m in terms of $(E_n^m/E_n)E_n$ or E_n-E_n^{im} to make E_n's as primary dependent variables, equation (3.6.9.1.1) is modified as

$$h\frac{\partial E_n}{\partial t}+\frac{\partial h}{\partial t}E_n+\mathbf{q}\cdot\nabla E_n-\nabla\cdot\left(h\mathbf{K}\cdot\nabla E_n\right)+\left(\nabla\cdot\mathbf{q}\right)\frac{E_n{}^m}{E_n}E_n$$
$$=\mathbf{q}\cdot\nabla E_n{}^{im}-\nabla\cdot\left(h\mathbf{K}\cdot\nabla E_n{}^{im}\right)M_{E_n{}^{as}}+M_{E_n{}^{rs}}+M_{E_n{}^{is}}+hR_{E_n} \quad \textbf{(3.6.9.1.2)}$$

To solve equation (3.6.9.1.2), assign

$$R_{HS}=0 \quad and \quad L_{HS}=\left(S_S+S_R+S_I-\partial h/\partial t\right)E_n{}^m/E_n \quad \textbf{(3.6.9.1.3)}$$

Then the right hand side R_{HS} and left hand side L_{HS} should be continuously calculated as following.

$$M_{E_n}{}^{rs}=\begin{cases}S_R*E_{n^{rs}}, & if\ S_R>0 \Rightarrow R_{HSn}=R_{HSn}+M_{E_n}{}^{rs}\\ S_R*E_n{}^m, & if\ S_R\le 0 \Rightarrow L_{HSn}=L_{HSn}-S_R\end{cases} \quad \textbf{(3.6.9.1.4)}$$

$$M_{E_n}{}^{as}=\begin{cases}S_S*E_{n^{as}}, & if\ S_S>0 \Rightarrow R_{HSn}=R_{HSn}+M_{E_n}{}^{as},\\ S_S*E_n{}^m, & if\ S_S\le 0 \Rightarrow L_{HSn}=L_{HSn}-S_S\end{cases} \quad \textbf{(3.6.9.1.5)}$$

$$M_{E_n}{}^{is}=\begin{cases}S_I*E_{n^{is}}{}^m, & if\ S_I>0 \Rightarrow R_{HSn}=R_{HSn}+M_{E_n}{}^{is}\\ S_I*E_n{}^m, & if\ S_I\le 0 \Rightarrow L_{HSn}=L_{HSn}-S_I\end{cases} \quad \textbf{(3.6.9.1.6)}$$

Equation (3.6.8.1.1) is then simplified as:

$$h\frac{\partial E_n}{\partial t}+\frac{\partial h}{\partial t}E_n+\mathbf{q}\cdot\nabla E_n-\nabla\cdot(h\mathbf{K}\cdot\nabla E_n)+L_{HS}E_n=\mathbf{q}\cdot\nabla E_n{}^{im}-\nabla\cdot\left(h\mathbf{K}\cdot\nabla E_n{}^{im}\right)+R_{HS}+hR_{E_n} \quad \textbf{(3.6.9.1.7)}$$

Assign the true transport velocity $\mathbf{v}_{\text{true}}$ as follows

$$h\mathbf{v}_{true}=\mathbf{q} \quad \textbf{(3.6.9.1.8)}$$

Equation (3.6.9.1.7) in the Lagrangian and Eulerian form is written as follows. In Lagrangian step,

$$h\frac{dE_n}{d\tau}=h\frac{\partial E_n}{\partial t}+\mathbf{q}\cdot\nabla E_n=0 \ \Rightarrow\ \frac{dE_n}{d\tau}=\frac{\partial E_n}{\partial t}+\mathbf{v}_{\text{true}}\cdot\nabla E_n=0 \quad \textbf{(3.6.9.1.9)}$$

In Eulerian step,

$$h\frac{dE_n}{d\tau}-\nabla\cdot(h\mathbf{K}\cdot\nabla E_n)+\left(L_{HS}+\frac{\partial h}{\partial t}\right)E_n=\mathbf{q}\cdot\nabla E_n{}^{im}-\nabla\cdot\left(h\mathbf{K}\cdot\nabla E_n{}^{im}\right)+R_{HS}+hR_{E_n} \quad \textbf{(3.6.9.1.10)}$$

Equation (3.6.9.1.10) written in a slightly different form is shown as

$$\frac{dE_n}{d\tau}-D+KE_n=T+R_L \quad \textbf{(3.6.9.1.11)}$$

where

$$D=\frac{1}{h}\nabla\cdot(h\mathbf{K}\cdot\nabla E_n) \quad \textbf{(3.6.9.1.12)}$$

$$K = \frac{\left(L_{HS} + \frac{\partial h}{\partial t}\right)}{h} \quad \textbf{(3.6.9.1.13)}$$

$$R_L = \frac{R_{HS} + hR_{E_n}}{h} \quad \textbf{(3.6.9.1.14)}$$

$$T = \frac{1}{h}\left[\mathbf{q} \cdot \nabla E_n^{\ im} - \nabla \cdot \left(h\mathbf{K} \cdot \nabla E_n^{\ im}\right)\right] \quad \textbf{(3.6.9.1.15)}$$

According to section 3.6.4,

$$[A1]\{D\} = -[A2]\{E_n\} + \{B1\} \quad \textbf{(3.6.9.1.16)}$$

where

$$A1_{ij} = \int_R N_i h N_j dR \quad \textbf{(3.6.9.1.17)}$$

$$A2_{ij} = \int_R \nabla N_i \cdot (h\mathbf{K} \cdot \nabla N_j) dR \quad \textbf{(3.6.9.1.18)}$$

$$B1_i = \int_B \mathbf{n} N_i \cdot (h\mathbf{K} \cdot \nabla E_n) dB \quad \textbf{(3.6.9.1.19)}$$

Lump matrix [A1] into diagonal matrix and assign

$$QE_{ij} = A2_{ij} / A1_{ii} \quad \textbf{(3.6.9.1.20)}$$

$$QB1_i = B1_i / A1_{ii} \quad \textbf{(3.6.9.1.21)}$$

Then

$$\{D\} = \{D1\} + \{QB1\} \quad \textbf{(3.6.9.1.22)}$$

where

$$\{D1\} = -[QE]\{E_n\} \quad \textbf{(3.6.9.1.23)}$$

Approximate T by a linear combination of the base functions as follows:

$$T \approx \hat{T} = \sum_{j=1}^{N} T_j(t) N_j(R) \quad \textbf{(3.6.9.1.24)}$$

According to equation (3.6.9.1.24), the integration of equation (3.6.9.1.15) can be written as

$$\int_R N_i hT dR = \int_R N_i h \sum_{j=1}^{N} T_j(t) N_j(R) dR = \int_R N_i \left[\mathbf{q} \cdot \nabla E_n^{\ im} - \nabla \cdot \left(h\mathbf{K} \cdot \nabla E_n^{\ im}\right)\right] dR \quad \textbf{(3.6.9.1.25)}$$

Further, we obtain

$$\sum_{j=1}^{N}\left[\left(\int_R N_i h N_j dR\right) T_j\right] = \int_R N_i \mathbf{q} \cdot \nabla E_n^{\ im} dR + \int_R \nabla N_i \cdot \left(h\mathbf{K} \cdot \nabla E_n^{\ im}\right) dR - \int_B \mathbf{n} \cdot N_i \left(h\mathbf{K} \cdot \nabla E_n^{\ im}\right) dB \quad \textbf{(3.6.9.1.26)}$$

Approximate E_n^{im} by a linear combination of the base functions as follows:

$$E_n^{im} \approx \hat{E}_n^{im} = \sum_{j=1}^{N} E_{nj}^{im}(t) N_j(R) \qquad \textbf{(3.6.9.1.27)}$$

Equation (3.6.9.1.26) is further expressed as

$$\begin{aligned} \sum_{j=1}^{N}\left[\left(\int_R N_i h N_j dR\right) T_j\right] &= \sum_{j=1}^{N}\left[\left(\int_R N_i \mathbf{q}\cdot\nabla N_j dR\right)(E_n^{im})_j\right] \\ &+\sum_{j=1}^{N}\left[\left(\int_R \nabla N_i\cdot\left(h\mathbf{K}\cdot\nabla N_j\right) dR\right)(E_n^{im})_j\right] - \int_B \mathbf{n}\cdot N_i\left(h\mathbf{K}\cdot\nabla E_n^{im}\right) dB \end{aligned} \qquad \textbf{(3.6.9.1.28)}$$

Assign matrices [A3], and load vector {B2} as following

$$A3_{ij} = \int_R N_i \mathbf{q}\cdot\nabla N_j dR \qquad \textbf{(3.6.9.1.29)}$$

$$B2_i = -\int_B \mathbf{n}\cdot N_i\left(h\mathbf{K}\cdot\nabla E_n^{im}\right) dB \qquad \textbf{(3.6.9.1.30)}$$

Assign

$$QT_{ij} = (A2_{ij} + A3_{ij}) / A1_{ii} \qquad \textbf{(3.6.9.1.31)}$$

$$QB2_i = B2_i / A1_{ii} \qquad \textbf{(3.6.9.1.32)}$$

Equation (3.6.9.1.28) is expressed as

$$\{T\} = \{T1\} + \{QB2\} \qquad \textbf{(3.6.9.1.33)}$$

where

$$\{T1\} = [QT]\{E_n^{im}\} \qquad \textbf{(3.6.9.1.34)}$$

So that equation (3.6.9.1.11) is then expressed as

$$\frac{dE_n}{d\tau} - D1 + KE_n = T1 + R_L + B \qquad \textbf{(3.6.9.1.35)}$$

where B=B1+B2. For boundary node i = b, the boundary term {B} should be calculated as follows.

For Dirichlet boundary condition

$$E_n^m = E_n^m(x_b, y_b, t) \;\Rightarrow\; B_i = \int_B \mathbf{n}\cdot N_i\left(h\mathbf{K}\cdot\nabla E_n^m\right) dB \Big/ A1_{ii} \qquad \textbf{(3.6.9.1.36)}$$

Variable boundary condition

< Case 1 > when flow is going in from outside (**n·q** < 0)

$$\begin{aligned} &\mathbf{n}\cdot\left(\mathbf{q}E_n^m - h\mathbf{K}\cdot\nabla E_n^m\right) = \mathbf{n}\cdot\mathbf{q}E_n^m(x_b, y_b, t) \;\Rightarrow \\ &B_i = \int_B \mathbf{n}N_i\cdot\mathbf{q}E_n^m dB \Big/ A1_{ii} - \int_B \mathbf{n}N_i\cdot\mathbf{q}E_n^m(x_b, y_b, t) dB \Big/ A1_{ii} \end{aligned} \qquad \textbf{(3.6.9.1.37)}$$

< Case 2 > Flow is going out from inside (**n·q** > 0):

$$-\mathbf{n}\cdot\left[h\mathbf{K}\cdot\nabla E_n{}^m(x_b,y_b,t)\right]=0 \quad\Rightarrow\quad B_i=0 \tag{3.6.9.1.38}$$

Cauchy boundary condition

$$\begin{aligned}&\mathbf{n}\cdot\left[\mathbf{q}E_n{}^m(x_b,y_b,t)-h\mathbf{K}\cdot\nabla E_n{}^m(x_b,y_b,t)\right]=q_b(t)\\ &\Rightarrow\quad B_i=\int_B N_i\left[\mathbf{n}\cdot\mathbf{q}E_n{}^m(x_b,y_b,t)-q_b(t)\right]dB\Big/QA_{ii}\\ &=\sum_{j=1}^{N}\left[\left(\int_B N_i\mathbf{n}\cdot\mathbf{q}N_j dB\right)E_{nj}{}^m(t)\right]\Big/QA_{ii}-\left(\int_B N_i dB\right)B\Big/QA_{ii}\end{aligned} \tag{3.6.9.1.39}$$

Neumann boundary condition

$$-\mathbf{n}\cdot\left[h\mathbf{K}\cdot\nabla E_n{}^m(x_b,y_b,t)\right]=q_b(t)\quad\Rightarrow\quad B_i=-\int_B N_i q_b(t)dB\Big/QA_{ii}=-\left(\int_B N_i dB\right)B\Big/QA_{ii} \tag{3.6.9.1.40}$$

River/stream-overland interface boundary condition

$$\begin{aligned}&\mathbf{n}\cdot\left[\mathbf{q}E_n{}^m(x_b,y_b,t)-h\mathbf{K}\cdot\nabla E_n{}^m(x_b,y_b,t)\right]=q_b(h_b(t))\quad\Rightarrow\\ &B_i=\int_B N_i\left[\mathbf{n}\cdot\mathbf{q}E_n{}^m(x_b,y_b,t)-q_b(h_b(t))\right]dB\Big/QA_{ii}\\ &=\sum_{j=1}^{N}\left[\left(\int_B N_i\mathbf{n}\cdot\mathbf{q}N_j dB\right)E_{nj}{}^m(t)\right]\Big/QA_{ii}-\left(\int_B N_i dB\right)B\Big/QA_{ii}\end{aligned} \tag{3.6.9.1.41}$$

Equation (3.6.9.1.35) written in matrix form is then expressed as

$$\begin{aligned}&\frac{[U]}{\Delta\tau}\left(\{E_n\}-\{E_n{}^*\}\right)-W_1\{D1\}-W_2\{D1^*\}+W_1\{K\}^T[U]\{E_n\}+W_2\{K^*\}^T[U]\{E_n{}^*\}\\ &=W_1\{T1\}+W_2\{T1^*\}+W_1\{RL\}+W_2\{RL^*\}+W_1\{B\}+W_2\{B^*\}\end{aligned} \tag{3.6.9.1.42}$$

At upstream flux boundary nodes, equation (3.6.9.1.42) cannot be applied because Δτ equals zero. Thus, we propose a modified LE approach in which the matrix equation for upstream boundary nodes is obtained by explicitly applying the finite element method to the boundary conditions. For example, at the upstream variable boundary

$$\int_B N_i n\cdot(qE_n{}^m-hK\cdot\nabla E_n{}^m)dB=\int_B N_i n\cdot qE_n{}^m(x_b,y_b,t)dB \tag{3.6.9.1.43}$$

So that the following matrix equation can be assembled at the boundary nodes

$$[QF]\{E_n{}^m\}=[QB]\{B\} \tag{3.6.9.1.44}$$

in which

$$QF_{ij}=\int_B(N_i\mathbf{n}\cdot\mathbf{q}N_j-N_i\mathbf{n}\cdot h\mathbf{K}\cdot\nabla N_j)dB \tag{3.6.9.1.45}$$

$$QB_{ij} = \int_B N_i \mathbf{n} \cdot \mathbf{q} N_j dB \tag{3.6.9.1.46}$$

$$B_i = E_n{}^m (x_b, y_b, t) \tag{3.6.9.1.47}$$

3.6.9.2 Mixed Predictor-corrector/Operator-splitting scheme

Recall the simplified equation for 2-D kinetic variable transport mixed Predictor-corrector/Operator-splitting scheme transport step in advection form, equation (3.6.8.2.2), as follows

$$h\frac{\partial E_n{}^m}{\partial t} + \mathbf{q} \cdot \nabla E_n{}^m - \nabla \cdot (h\mathbf{K} \cdot \nabla E_n{}^m) + \left(L_{HS} + \frac{\partial h}{\partial t} \right) E_n{}^m = R_{HS} + hR_{E_n}{}^n - \frac{\partial h}{\partial t} (E_n^{im})^n \tag{3.6.9.2.1}$$

Assign the true transport velocity $\mathbf{v}_{\text{true}}$ as follows

$$h\mathbf{v}_{true} = \mathbf{q} = W_1 \mathbf{q}^{n+1} + W_2 \mathbf{q}^n \tag{3.6.9.2.2}$$

Equation (3.6.9.2.1) in the Lagrangian and Eulerian form is written as follows. In lagrangian step,

$$h\frac{dE_n{}^m}{d\tau} = h\frac{\partial E_n{}^m}{\partial t} + \mathbf{q} \cdot \nabla E_n{}^m = 0 \quad \Rightarrow \quad \frac{\partial E_n{}^m}{\partial t} + \mathbf{v}_{true} \cdot \nabla E_n{}^m = 0 \tag{3.6.9.2.3}$$

In Eulerian step,

$$h\frac{dE_n{}^m}{d\tau} - \nabla \cdot (h\mathbf{K} \cdot \nabla E_n{}^m) + \left(L_{HS} + \frac{\partial h}{\partial t} \right) E_n{}^m = R_{HS} + hR_{E_n}{}^n - \frac{\partial h}{\partial t} (E_n^{im})^n \tag{3.6.9.2.4}$$

Equation (3.6.9.3.4) written in a slightly different form is shown as

$$\frac{dE_n{}^m}{d\tau} - D + K * E_n{}^m = R_L \tag{3.6.9.2.5}$$

where

$$D = \frac{1}{h} \nabla \cdot (h\mathbf{K} \cdot \nabla E_n{}^m) \tag{3.6.9.2.6}$$

$$K = \frac{\left(L_{HS} + \frac{\partial h}{\partial t} \right)}{h} \tag{3.6.9.2.7}$$

$$R_L = \frac{R_{HS} + hR_{E_n}{}^n - \frac{\partial h}{\partial t} (E_n^{im})^n}{h} \tag{3.6.9.2.8}$$

Equation (3.6.9.2.5) written in matrix form is then expressed as

$$\begin{aligned} &\frac{[\mathrm{U}]}{\Delta \tau} \left(\{\mathrm{E_n}^{\mathrm{m}}\} - \{\mathrm{E_n}^{\mathrm{m}*}\} \right) - \mathrm{W_1}\{\mathrm{D}\} - \mathrm{W_2}\{\mathrm{D}^*\} + \mathrm{W_1}\{\mathrm{K}\}^{\mathrm{T}} [\mathrm{U}] \{\mathrm{E_n}^{\mathrm{m}}\} + \mathrm{W_2}\left\{\mathrm{K}^*\right\}^{\mathrm{T}} [\mathrm{U}] \{\mathrm{E_n}^{\mathrm{m}*}\} \\ &= \mathrm{W_1}\{\mathrm{R_L}\} + \mathrm{W_2}\left\{(\mathrm{R_L})^*\right\} \end{aligned} \tag{3.6.9.2.9}$$

Same as that in section 3.6.9.1,

$$\{D\} = -[QD]\{E_n^{\ m}\} + \{QB\} \qquad \textbf{(3.6.9.2.10)}$$

At upstream flux boundary nodes, equation (3.6.9.2.9) cannot be applied because $\Delta\tau$ equals zero. Thus, we propose a modified LE approach in which the matrix equation for upstream boundary nodes is obtained by explicitly applying the finite element method to the boundary conditions.

3.6.9.3 Operator-splitting scheme

Equation (3.6.8.3.2) can be solved through the same procedure as that in section 3.6.9.2, except that

$$R_L = \frac{R_{HS}}{h} \qquad \textbf{(3.6.9.3.1)}$$

3.6.10 Application of the Lagrangian-Eulerian Approach for All Interior Nodes and Downstream Boundary Nodes with the Finite Element Method Applied to the Conservative Form of the Transport Equations for the Upstream Flux Boundaries to Solve 2-D Kinetic Variable Transport

3.6.10.1 Fully-Implicit Scheme

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.6.9.1, and the matrix equation for upstream boundary nodes is obtained through the same procedure as that in section 3.6.7.1.

3.6.10.2 Mixed Predictor-Corrector and Operator-Splitting Method

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.6.9.2, and the matrix equation for upstream boundary nodes is obtained through the same procedure as that in section 3.6.7.2.

3.6.10.3 Operator-Splitting Approach

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.6.9.3, and the matrix equation for upstream boundary nodes is obtained through the same procedure as that in section 3.6.7.3.

3.6.11 Application of the Lagrangian-Eulerian Approach for All Interior Nodes and Downstream Boundary Nodes with the Finite Element Method Applied to the Advective Form of the Transport Equations for the Upstream Flux Boundaries to Solve 2-D Kinetic Variable Transport

3.6.11.1 Fully-Implicit Scheme

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.6.9.1, and the matrix equation for upstream boundary nodes is obtained through the same procedure as that in section 3.6.8.1.

3.6.11.2 Mixed Predictor-Corrector and Operator-Splitting Method

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.6.9.2, and the matrix equation for upstream boundary nodes is obtained through the same procedure as that in section 3.6.8.2.

3.6.11.3 Operator-Splitting Approach

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.6.9.3, and the matrix equation for upstream boundary nodes is obtained through the same procedure as that in section 3.6.8.3.

3.7 Solving Three-Dimensional Subsurface Water Quality Transport Equations

In this section, we present the numerical approaches employed to solve the governing equations of reactive chemical transport. Ideally, one would like to use a numerical approach that is accurate, efficient, and robust. Depending on the specific problem at hand, different numerical approaches may be more suitable. For research applications, accuracy is a primary requirement, because one does not want to distort physics due to numerical errors. On the other hand, for large field-scale problems, efficiency and robustness are primary concerns as long as accuracy remains within the bounds of uncertainty associated with model parameters. Thus, to provide accuracy for research applications and efficiency and robustness for practical applications, three coupling strategies were investigated to deal with reactive chemistry. They are: (1) a fully-implicit scheme, (2) a mixed predictor-corrector/operator-splitting method, and (3) an operator-splitting method. For each time-step, we first solve the advective-dispersive transport equation with or without reaction terms, kinetic-variable by kinetic-variable. We then solve the reactive chemical system node-by-node to yield concentrations of all species.

Five numerical options are provided to solve the advective-dispersive transport equations: Option 1- application of the Finite Element Method (FEM) to the conservative form of the transport equations, Option 2 - application of the FEM to the advective form of the transport equations, Option 3 - application of the modified Lagrangian-Eulerian (LE) approach to the Largrangian form of the transport equations, Option 4 - LE approach for all interior nodes and downstream boundary nodes with the FEM applied to the conservative form of the transport equations for the upstream flux boundaries, and Option 5 - LE approach for all interior and downstream boundary nodes with the FEM applied to the advective form of the transport equations for upstream flux boundaries.

3.7.1 Application of the Finite Element Method to the Conservative Form of the Reactive Chemical Transport Equations

3.7.1.1 Fully-Implicit Scheme

Assign the right-hand side term R_{HS} and left hand side term L_{HS} as follows.

$$\begin{aligned} &If\ q \le 0, \quad M_{E_n}{}^{as} = qE_n{}^m, \quad L_{HS} = -q, \quad R_{HS} = 0 \\ &Else\ q > 0, \quad M_{E_n}{}^{as} = qE_{n^{as}}, \quad L_{HS} = 0, \quad R_{HS} = M_{E_n}{}^{as} \end{aligned} \qquad \textbf{(3.7.1.1.1)}$$

Then equation (2.7.22) is modified as

$$\theta \frac{\partial E_n}{\partial t} + \frac{\partial \theta}{\partial t} E_n + \nabla \cdot (\mathbf{V} E_n{}^m) - \nabla \cdot (\theta \mathbf{D} \cdot \nabla E_n{}^m) + L_{HS} E_n{}^m = R_{HS} + \theta R_{E_n} \qquad \textbf{(3.7.1.1.2)}$$

According to the fully-implicit scheme, equation (3.7.1.1.2) can be separated into two equations as follows.

$$\theta \frac{E_n{}^{n+1/2} - E_n{}^n}{\Delta t} + \frac{\partial \theta}{\partial t} E_n + \nabla \cdot (\mathbf{V} E_n{}^m) - \nabla \cdot (\theta \mathbf{D} \cdot \nabla E_n{}^m) + L_{HS} E_n{}^m = R_{HS} + \theta R_{E_n} \qquad \textbf{(3.7.1.1.3)}$$

$$\frac{E_n{}^{n+1} - E_n{}^{n+1/2}}{\Delta t} = 0 \qquad \textbf{(3.7.1.1.4)}$$

First, we express $E_n{}^m$ in terms of $(E_n{}^m/E_n)\cdot E_n$ or $(E_n–E_n{}^{im})$ to make E_n's as primary dependent variables, so that $E_n{}^{n+1/2}$ can be solved through equation (3.7.1.1.3). It is noted that the approach of expressing $E_n{}^m$ in terms of $(E_n{}^m/E_n)\cdot E_n$ improves model accuracy but is less robust than the approach of expressing $E_n{}^m$ in terms of $(E_n–E_n{}^{im})$ taken in Yeh et al. [2004]. Second, we solve equation (3.7.1.1.4) together with algebraic equations for equilibrium reactions using BIOGEOCHEM [Fang et al., 2003] to obtain all individual species concentrations. Iteration between these two steps is needed because the new reaction terms $RA_n{}^{n+1}$ and the equation coefficients in equation (3.7.1.1.3) need to be updated by the calculation results of (3.7.1.1.4). To improve the standard SIA method, the nonlinear reaction terms are approximated by the Newton-Raphson linearization.

Option 1: Express $E_n{}^m$ in terms of $(E_n{}^m/E_n)\ E_n{}^m$

$$\begin{aligned} &\theta \frac{\partial E_n}{\partial t} + \nabla \cdot \left(\mathbf{V} \frac{E_n{}^m}{E_n} E_n \right) - \nabla \cdot \left(\theta \mathbf{D} \cdot \frac{E_n{}^m}{E_n} \nabla E_n \right) \\ &- \nabla \cdot \left[\theta \mathbf{D} \cdot \left(\nabla \frac{E_n{}^m}{E_n} \right) E_n \right] + \left(L_{HS} \frac{E_n{}^m}{E_n} + \frac{\partial \theta}{\partial t} \right) E_n = R_{HS} + \theta R_{E_n} \end{aligned} \qquad \textbf{(3.7.1.1.5)}$$

Use Galerkin or Petrov-Galerkin Finite-Element Method for the spatial descretization of transport equation: choose weighting function identical to base function. For Petriov-Galerkin method, apply weighting function one-order higher than the base function to advection term. Integrate equation (3.7.1.1.5) in the spatial dimensions over the entire region as follows.

$$\int_R N_i \left[\theta \frac{\partial E_n}{\partial t} - \nabla \cdot \left(\theta \mathbf{D} \cdot \frac{E_n{}^m}{E_n} \nabla E_n \right) + \left(L_{HS} \frac{E_n{}^m}{E_n} + \frac{\partial \theta}{\partial t} \right) E_n \right] dR$$
$$+ \int_R W_i \left\{ \nabla \cdot \left(\mathbf{V} \frac{E_n{}^m}{E_n} E_n \right) - \nabla \cdot \left[\theta \mathbf{D} \cdot \left(\nabla \frac{E_n{}^m}{E_n} \right) E_n \right] \right\} dR = \int_R N_i (R_{HS} + \theta R_{E_n}) dR \tag{3.7.1.1.6}$$

Further, we obtain

$$\int_R N_i \theta \frac{\partial E_n}{\partial t} dR - \int_R \nabla W_i \cdot \mathbf{V} \frac{E_n{}^m}{E_n} E_n dR + \int_R \nabla N_i \cdot \left(\theta \mathbf{D} \cdot \frac{E_n{}^m}{E_n} \nabla E_n \right) dR$$
$$+ \int_R \nabla W_i \cdot \left[\theta \mathbf{D} \cdot \left(\nabla \frac{E_n{}^m}{E_n} \right) E_n \right] dR + \int_R N_i \left(L_{HS} \frac{E_n{}^m}{E_n} + \frac{\partial \theta}{\partial t} \right) E_n dR = \int_R N_i (R_{HS} + \theta R_{E_n}) dR \tag{3.7.1.1.7}$$
$$- \int_B \mathbf{n} \cdot W_i \mathbf{V} \frac{E_n{}^m}{E_n} E_n dB + \int_B \mathbf{n} \cdot \left(N_i \theta \mathbf{D} \cdot \frac{E_n{}^m}{E_n} \nabla E_n \right) dB + \int_B \mathbf{n} \cdot \left[W_i \theta \mathbf{D} \cdot \left(\nabla \frac{E_n{}^m}{E_n} \right) E_n \right] dB$$

Approximate solution E_n by a linear combination of the base functions as follows.

$$E_n \approx \hat{E}_n = \sum_{j=1}^{N} E_{nj}(t) N_j(R) \tag{3.7.1.1.8}$$

Substituting equation (3.7.1.1.8) into equation (3.7.1.1.7), we obtain

$$\sum_{j=1}^{N} \left[\left(\int_R N_i \theta N_j dR \right) \frac{\partial E_{nj}(t)}{\partial t} \right]$$
$$+ \sum_{j=1}^{N} \left\{ \left\{ - \int_R \nabla W_i \cdot \mathbf{V} \frac{E_n{}^m}{E_n} N_j dR + \int_R \nabla W_i \cdot \left[\theta \mathbf{D} \cdot \left(\nabla \frac{E_n{}^m}{E_n} \right) N_j \right] dR \right\} E_{nj}(t) \right\}$$
$$+ \sum_{j=1}^{N} \left\{ \left[\int_R \nabla N_i \cdot \left(\theta \mathbf{D} \cdot \frac{E_n{}^m}{E_n} \nabla N_j \right) dR + \int_R N_i \left(L_{HS} \frac{E_n{}^m}{E_n} + \frac{\partial \theta}{\partial t} \right) N_j dR \right] E_{nj}(t) \right\} \tag{3.7.1.1.9}$$
$$= \int_R N_i (R_{HS} + \theta R_{E_n}) dR - \int_B \mathbf{n} \cdot W_i \mathbf{V} E_n{}^m dB + \int_B \mathbf{n} \cdot \left(N_i \theta \mathbf{D} \cdot \nabla E_n{}^m \right) dB$$

Equation (3.7.1.1.9) can be written in matrix form as

$$[Q1] \left\{ \frac{\partial E_n}{\partial t} \right\} + [Q2]\{E_n\} + [Q3]\{E_n\} = \{RLS\} + \{B\} \tag{3.7.1.1.10}$$

where the matrices [Q1], [Q2], [Q3] and load vectors {RLS}, and {B} are given by

$$Q1_{ij} = \int_R N_i \theta N_j dR \tag{3.7.1.1.11}$$

$$Q2_{ij} = -\int_R \nabla W_i \cdot \mathbf{V} \frac{E_n{}^m}{E_n} N_j dR + \int_R \nabla W_i \cdot \left[\theta \mathbf{D} \cdot \left(\nabla \frac{E_n{}^m}{E_n} \right) N_j \right] dR \qquad \textbf{(3.7.1.1.12)}$$

$$Q3_{ij} = \int_R \nabla N_i \cdot \left(\theta \mathbf{D} \cdot \frac{E_n{}^m}{E_n} \nabla N_j \right) dR + \int_R N_i \left(L_{HS} \frac{E_n{}^m}{E_n} + \frac{\partial \theta}{\partial t} \right) N_j dR \qquad \textbf{(3.7.1.1.13)}$$

$$RLS_i = \int_R N_i (R_{HS} + \theta R_{E_n}) dR \qquad \textbf{(3.7.1.1.14)}$$

$$B_i = -\int_B \mathbf{n} \cdot W_i \mathbf{V} E_n{}^m dB + \int_B \mathbf{n} \cdot \left(N_i \theta \mathbf{D} \cdot \nabla E_n{}^m \right) dB \qquad \textbf{(3.7.1.1.15)}$$

At n+1-th time step, equation (3.7.1.1.10) is approximated as

$$\begin{aligned} &[Q1] \frac{\left\{ E_n{}^{n+1/2} \right\} - \left\{ E_n{}^n \right\}}{\Delta t} + W_{V1}[Q2^{n+1}] \left\{ E_n{}^{n+1/2} \right\} + W_{V2}[Q2^n] \left\{ E_n{}^n \right\} \\ &+ W_1[Q3^{n+1}] \left\{ E_n{}^{n+1/2} \right\} + W_2[Q3^n] \left\{ E_n{}^n \right\} \\ &= W_1\{RLS^{n+1}\} + W_2\{RLS^n\} + W_1\{B^{n+1}\} + W_2\{B^n\} \end{aligned} \qquad \textbf{(3.7.1.1.16)}$$

where W_{V1}, W_{V2}, W_1 and W_2 are time weighting factors, matrices and vectors with superscripts $^{n+1}$ and ^{n} are evaluated over the region at the new time step n+1 and at the old time step n, respectively.

So that

$$\begin{aligned} &\left(\frac{[Q1]}{\Delta t} + W_{V1}[Q2^{n+1}] + W_1[Q3^{n+1}] \right) \left\{ E_n{}^{n+1/2} \right\} \\ &= \left(\frac{[Q1]}{\Delta t} - W_{V2}[Q2^n] - W_2[Q3^n] \right) \left\{ E_n{}^n \right\} + W_1\{SS^{n+1}\} + W_2\{SS^n\} + W_1\{B^{n+1}\} + W_2\{B^n\} \end{aligned} \qquad \textbf{(3.7.1.1.17)}$$

Option 2: Express $E_n{}^m$ in terms of E_n-$E_n{}^{im}$

Use Galerkin or Petrov-Galerkin Finite-Element Method for the spatial descretization of transport equation. Integrate equation (3.7.1.1.3) in the spatial dimensions over the entire region as follows.

$$\begin{aligned} &\int_R N_i \left[\theta \frac{\partial E_n}{\partial t} + \frac{\partial \theta}{\partial t} E_n - \nabla \cdot \left(\theta \mathbf{D} \cdot E_n{}^m \right) + L_{HS} E_n{}^m \right] dR + \int_R W_i \left[\nabla \cdot \left(\mathbf{V} E_n{}^m \right) \right] dR \\ &= \int_R N_i (R_{HS} + \theta R_{E_n}) dR \end{aligned} \qquad \textbf{(3.7.1.1.18)}$$

Further, we obtain

$$\int_R N_i \left(\theta \frac{\partial E_n}{\partial t} + \frac{\partial \theta}{\partial t} E_n \right) dR - \int_R \nabla W_i \cdot \mathbf{V} E_n{}^m dR + \int_R \nabla N_i \cdot \left(\theta \mathbf{D} \cdot \nabla E_n{}^m \right) dR + \int_R N_i L_{HS} E_n{}^m dR$$
$$= \int_R N_i (R_{HS} + \theta R_{E_n}) dR - \int_B \mathbf{n} \cdot W_i \mathbf{V} E_n{}^m dB + \int_B \mathbf{n} \cdot \left(N_i \theta \mathbf{D} \cdot \nabla E_n{}^m \right) dB \qquad \textbf{(3.7.1.1.19)}$$

Approximate solution E_n by a linear combination of the base functions as equation (3.7.1.1.8). Substituting equation (3.7.1.1.8) into equation (3.7.1.1.19), we obtain

$$\sum_{j=1}^{N} \left[\left(\int_R N_i \theta^n N_j dR \right) \frac{\partial E_{nj}(t)}{\partial t} \right] + \sum_{j=1}^{N} \left[\left(-\int_R \nabla W_i \cdot \mathbf{V} N_j dR \right) E_{nj}{}^m(t) \right] +$$
$$\sum_{j=1}^{N} \left[\left(\int_R N_i \frac{\partial \theta}{\partial t} N_j dR \right) E_{nj}(t) \right] + \sum_{j=1}^{N} \left\{ \left[\int_R \nabla N_i \cdot \left(\theta \mathbf{D} \cdot \nabla N_j \right) dR + \int_R N_i L_{HS} N_j dR \right] E_{nj}{}^m(t) \right\} \qquad \textbf{(3.7.1.1.20)}$$
$$= \int_R N_i (R_{HS} + \theta R_{E_n}) dR - \int_B \mathbf{n} \cdot W_i \mathbf{V} E_n{}^m dB + \int_B \mathbf{n} \cdot \left(N_i \theta \mathbf{D} \cdot \nabla E_n{}^m \right) dB$$

Equation (3.7.1.1.20) can be written in matrix form as

$$[Q1] \left\{ \frac{\partial E_n}{\partial t} \right\} + [Q4]\{E_n\} + [Q2]\{E_n{}^m\} + [Q3]\{E_n{}^m\} = \{RLS\} + \{B\} \qquad \textbf{(3.7.1.1.21)}$$

where the matrices [Q1], [Q4], [Q2], [Q3] and load vectors {RLS}, and {B} are given by

$$Q1_{ij} = \int_R N_i \theta N_j dR, \qquad Q4_{ij} = \int_R N_i \frac{\partial \theta}{\partial t} N_j dR \qquad \textbf{(3.7.1.1.22)}$$

$$Q2_{ij} = -\int_R \nabla W_i \cdot \mathbf{V} N_j dR \qquad \textbf{(3.7.1.1.23)}$$

$$Q3_{ij} = \int_R \nabla N_i \cdot \left(\theta \mathbf{D} \cdot \nabla N_j \right) dR + \int_R N_i L_{HS} N_j dR \qquad \textbf{(3.7.1.1.24)}$$

$$RLS_i = \int_R N_i (R_{HS} + \theta R_{E_n}) dR \qquad \textbf{(3.7.1.1.25)}$$

$$B_i = -\int_B \mathbf{n} \cdot W_i \mathbf{V} E_n{}^m dB + \int_B \mathbf{n} \cdot \left(N_i \theta \mathbf{D} \cdot \nabla E_n{}^m \right) dB \qquad \textbf{(3.7.1.1.26)}$$

Express $E_n{}^m$ in terms of E_n-$E_n{}^{im}$, equation (3.7.1.1.21) is modified as

$$[Q1] \left\{ \frac{\partial E_n}{\partial t} \right\} + [Q4]\{E_n\} + [Q2]\{E_n\} + [Q3]\{E_n\} = [Q2]\{E_n{}^{im}\} + [Q3]\{E_n{}^{im}\}$$
$$+\{RLS\} + \{B\} \qquad \textbf{(3.7.1.1.27)}$$

At n+1-th time step, equation (3.7.1.1.27) is approximated as

$$[Q1]\frac{\left\{E_n^{\ n+1/2}\right\}-\left\{E_n^{\ n}\right\}}{\Delta t}+[Q4]\left\{E_n^{\ n+1/2}\right\}+W_{V1}[Q2^{n+1}]\left\{E_n^{\ n+1/2}\right\}+W_{V2}[Q2^n]\left\{E_n^{\ n}\right\}$$
$$+W_1[Q3^{n+1}]\left\{E_n^{\ n+1/2}\right\}+W_2[Q3^n]\left\{E_n^{\ n}\right\}=W_{V1}[Q2^{n+1}]\left\{\left(E_n^{\ im}\right)^{n+1/2}\right\}$$
$$+W_{V2}[Q2^n]\left\{\left(E_n^{\ im}\right)^n\right\}+W_1[Q3^{n+1}]\left\{\left(E_n^{\ im}\right)^{n+1/2}\right\}+W_2[Q3^n]\left\{\left(E_n^{\ im}\right)^n\right\}$$
$$+W_1\{RLS^{n+1}\}+W_2\{RLS^n\}+W_1\{B^{n+1}\}+W_2\{B^n\} \quad \textbf{(3.7.1.1.28)}$$

So that

$$\left(\frac{[Q1]}{\Delta t}+[Q4]+W_{V1}[Q2^{n+1}]+W_1[Q3^{n+1}]\right)\left\{E_n^{\ n+1/2}\right\}=\frac{[Q1]}{\Delta t}\left\{E_n^{\ n}\right\}-$$
$$\left(W_{V2}[Q2^n]+W_2[Q3^n]\right)*\left\{\left(E_n^{\ m}\right)^n\right\}+\left(W_{V1}[Q2^{n+1}]+W_1[Q3^{n+1}]\right)\left\{\left(E_n^{\ im}\right)^{n+1/2}\right\}+$$
$$W_1\{SS^{n+1}\}+W_2\{SS^n\}+W_1\{B^{n+1}\}+W_2\{B^n\} \quad \textbf{(3.7.1.1.29)}$$

For interior nodes i, B_i is zero, for boundary nodes i = b, B_i is calculated according to the specified boundary condition and shown as follows.

<u>Dirichlet boundary condition</u>

$$E_n^{\ m}=E_n^{\ m}(x_b,y_b,z_b,t) \quad \textbf{(3.7.1.1.30)}$$

<u>Variable boundary condition</u>

< Case 1 > when flow is going in from outside ($\mathbf{n}\cdot\mathbf{V}<0$)

$$\mathbf{n}\cdot(\mathbf{V}E_n^{\ m}-\theta\mathbf{D}\cdot\nabla E_n^{\ m})=\mathbf{n}\cdot\mathbf{V}E_n^{\ m}(x_b,y_b,z_b,t) \;\Rightarrow\; B_i=-\int_{\mathbf{B}}\mathbf{n}\cdot N_i\mathbf{V}E_n^{\ m}(x_b,y_b,z_b,t)dB \quad \textbf{(3.7.1.1.31)}$$

< Case 2 > Flow is going out from inside ($\mathbf{n}\cdot\mathbf{V}>0$):

$$-\mathbf{n}\cdot(\theta\mathbf{D}\cdot\nabla E_n^{\ m})=0 \;\Rightarrow\; B_i=-\int_B\mathbf{n}\cdot N_i\mathbf{V}E_n^{\ m}dB \quad \textbf{(3.7.1.1.32)}$$

<u>Cauchy boundary condition</u>

$$\mathbf{n}\cdot(\mathbf{V}E_n^{\ m}-\theta\mathbf{D}\cdot\nabla E_n^{\ m})=Q_{E_n^m}(x_b,y_b,z_b,t) \;\Rightarrow\; B_i=-\int_B N_iQ_{E_n^m}(x_b,y_b,z_b,t)dB \quad \textbf{(3.7.1.1.33)}$$

<u>Neumann boundary condition</u>

$$
\begin{aligned}
& -\mathbf{n}\cdot(\theta\mathbf{D}\cdot\nabla E_n{}^m) = Q_{E_n{}^m}(x_b, y_b, z_b, t) \\
& \Rightarrow \quad B_i = -\int_B \mathbf{n}\cdot N_i \mathbf{V} E_n{}^m dB - \int_B N_i Q_{E_n{}^m}(x_b, y_b, z_b, t) dB
\end{aligned}
\tag{3.7.1.1.34}
$$

River/stream-subsurface interface boundary condition

$$
\begin{aligned}
& \mathbf{n}\cdot(\mathbf{V}E_n{}^m - \theta\mathbf{D}\cdot\nabla E_n{}^m) \\
& = \frac{\mathbf{n}\cdot\mathbf{V}}{2}\left\{ [1+sign(\mathbf{n}\cdot\mathbf{V})]E_n{}^m + [1-sign(\mathbf{n}\cdot\mathbf{V})]\left(E_n{}^m\right)^{1D} \right\} \quad \Rightarrow \\
& B_i = -\int_B N_i \frac{\mathbf{n}\cdot\mathbf{V}}{2}\left\{ [1+sign(\mathbf{n}\cdot\mathbf{V})]E_n{}^m + [1-sign(\mathbf{n}\cdot\mathbf{V})]\left(E_n{}^m\right)^{1D} \right\} dB
\end{aligned}
\tag{3.7.1.1.35}
$$

Overland-subsurface interface boundary condition

$$
\begin{aligned}
& \mathbf{n}\cdot(\mathbf{V}E_n{}^m - \theta\mathbf{D}\cdot\nabla E_n{}^m) \\
& = \frac{\mathbf{n}\cdot\mathbf{V}}{2}\left\{ [1+sign(\mathbf{n}\cdot\mathbf{V})]E_n{}^m + [1-sign(\mathbf{n}\cdot\mathbf{V})]\left(E_n{}^m\right)^{2D} \right\} \quad \Rightarrow \\
& B_i = -\int_B N_i \frac{\mathbf{n}\cdot\mathbf{V}}{2}\left\{ [1+sign(\mathbf{n}\cdot\mathbf{V})]E_n{}^m + [1-sign(\mathbf{n}\cdot\mathbf{V})]\left(E_n{}^m\right)^{2D} \right\} dB
\end{aligned}
\tag{3.7.1.1.36}
$$

3.7.1.2 Mixed Predictor-Corrector and Operator-Splitting Method

According to the mixed predictor-corrector (on reaction rates) and operator-splitting (on immobile part of the kinetic variable) method, equation (3.7.1.1.2) can be separated into two equations as follows.

$$
\begin{aligned}
& \theta\frac{\left(E_n{}^m\right)^{n+1/2} - \left(E_n{}^m\right)^n}{\Delta t} + \frac{\partial\theta}{\partial t}E_n^m + \nabla\cdot(\mathbf{V}E_n{}^m) - \nabla\cdot(\theta\mathbf{D}\cdot\nabla E_n{}^m) \\
& + L_{HS}E_n{}^m = R_{HS} + \theta R_{E_n}{}^n - \frac{\partial\theta}{\partial t}(E_n^{im})^n
\end{aligned}
\tag{3.7.1.2.1}
$$

$$
\frac{E_n{}^{n+1} - [(E_n{}^m)^{n+1/2} + (E_n{}^{im})^{n+1/2}]}{\Delta t} = \theta R_{E_n}{}^{n+1} - \theta R_{E_n}{}^n - \frac{\partial \ell n\theta}{\partial t}(E_n^{im})^{n+1} + \frac{\partial \ell n\theta}{\partial t}(E_n^{im})^n
\tag{3.7.1.2.2}
$$

First, solve equation (3.7.1.2.1) and get $(E_n{}^m)^{n+1/2}$. Second, solve equation (3.7.1.2.2) together with algebraic equations representing equilibrium reactions using BIOGEOCHM scheme to obtain the individual species concentration.

Use Galerkin or Petrov-Galerkin Finite-Element Method for the spatial descretization of transport equation. Integrate equation (3.7.1.2.1) in the spatial dimensions over the entire region as follows.

$$\int_R N_i \left[\theta \frac{\partial E_n{}^m}{\partial t} - \nabla \cdot (\theta \mathbf{D} \cdot \nabla E_n{}^m) + \left(L_{HS} + \frac{\partial \theta}{\partial t} \right) E_n{}^m \right] dR + \int_R W_i \nabla \cdot \mathbf{V} E_n{}^m dR = \int_R N_i \left(R_{HS} + \theta R_{E_n}{}^n - \frac{\partial \theta}{\partial t} (E_n^{im})^n \right) dR \quad \textbf{(3.7.1.2.3)}$$

Further, we obtain

$$\int_R N_i \theta \frac{\partial E_n{}^m}{\partial t} dR - \int_R \nabla W_i \cdot \mathbf{V} E_n{}^m dR + \int_R \nabla N_i \cdot (\theta \mathbf{D} \cdot \nabla E_n{}^m) dR + \int_R N_i \left(L_{HS} + \frac{\partial \theta}{\partial t} \right) E_n{}^m dR = \int_R N_i \left(R_{HS} + \theta R_{E_n}{}^n - \frac{\partial \theta}{\partial t} (E_n^{im})^n \right) dR - \int_B \mathbf{n} \cdot W_i \mathbf{V} E_n{}^m dB + \int_B \mathbf{n} \cdot N_i (\theta \mathbf{D} \cdot \nabla E_n{}^m) dB \quad \textbf{(3.7.1.2.4)}$$

Approximate solution $E_n{}^m$ by a linear combination of the base functions as follows.

$$E_n{}^m \approx \hat{E}_n{}^m = \sum_{j=1}^{N} E_{nj}{}^m(t) N_j(R) \quad \textbf{(3.7.1.2.5)}$$

Substituting equation (3.7.1.2.5) into equation (3.7.1.2.4), we obtain

$$\sum_{j=1}^{N} \left[\left(\int_R N_i \theta N_j dR \right) \frac{\partial E_{nj}{}^m(t)}{\partial t} \right] - \sum_{j=1}^{N} \left[\left(\int_R \nabla W_i \cdot \mathbf{V} N_j dR \right) E_{nj}{}^m(t) \right] + \sum_{j=1}^{N} \left\{ \left[\int_R \nabla N_i \cdot (\theta \mathbf{D} \cdot \nabla N_j) dR + \int_R N_i \left(L_{HS} + \frac{\partial \theta}{\partial t} \right) N_j dR \right] E_{nj}{}^m(t) \right\} = \int_R N_i \left(R_{HS} + \theta R_{E_n}{}^n - \frac{\partial \theta}{\partial t} (E_n^{im})^n \right) dR - \int_B \mathbf{n} \cdot W_i \mathbf{V} E_n{}^m dB + \int_B \mathbf{n} \cdot N_i (\theta \mathbf{D} \cdot \nabla E_n{}^m) dB \quad \textbf{(3.7.1.2.6)}$$

Equation (3.7.1.2.6) can be written in matrix form as

$$[Q1] \left\{ \frac{dE_n{}^m}{dt} \right\} + [Q2] \left\{ E_n{}^m \right\} + [Q3] \left\{ E_n{}^m \right\} = \{RLS\} + \{B\} \quad \textbf{(3.7.1.2.7)}$$

where the matrices [Q1], [Q2], and [Q3], and load vectors {RLS} and {B} are given by

$$Q1_{ij} = \int_R N_i \theta N_j dR \quad \textbf{(3.7.1.2.8)}$$

$$Q2_{ij} = -\int_R \nabla W_i \cdot \mathbf{V} N_j dR \quad \textbf{(3.7.1.2.9)}$$

$$Q3_{ij} = \int_R \nabla N_i \cdot \left(\theta \mathbf{D} \cdot \nabla N_j \right) dR + \int_R N_i \left(L_{HS} + \frac{\partial \theta}{\partial t} \right) N_j dR \quad \textbf{(3.7.1.2.10)}$$

$$RLS_i = \int_R N_i \left(R_{HS} + \theta R_{E_n}{}^n - \frac{\partial \theta}{\partial t} (E_n^{im})^n \right) dR \qquad \textbf{(3.7.1.2.11)}$$

$$B_i = -\int_{\mathbf{B}} \mathbf{n} \cdot W_i \mathbf{V} E_n{}^m dB + \int_B \mathbf{n} \cdot (N_i \theta \mathbf{D} \cdot \nabla E_n{}^m) dB \qquad \textbf{(3.7.1.2.12)}$$

At n+1-th time step, equation (3.7.1.2.7) is approximated as

$$\begin{aligned} &[Q1] \frac{\left\{ \left(E_n{}^m\right)^{n+1/2} \right\} - \left\{ \left(E_n{}^m\right)^n \right\}}{\Delta t} + W_{V1}[Q2^{n+1}] \left\{ \left(E_n{}^m\right)^{n+1/2} \right\} + W_{V2}[Q2^n] \left\{ \left(E_n{}^m\right)^n \right\} \\ &+ W_1[Q3^{n+1}] \left\{ \left(E_n{}^m\right)^{n+1/2} \right\} + W_2[Q3^n] \left\{ \left(E_n{}^m\right)^n \right\} = W_1\{RLS^{n+1}\} + W_2\{RLS^n\} + W_1\{B^{n+1}\} + W_2\{B^n\} \end{aligned} \qquad \textbf{(3.7.1.2.13)}$$

So that

$$\begin{aligned} &\left(\frac{[Q1]}{\Delta t} + W_{V1}[Q2^{n+1}] + W_1[Q3^{n+1}] \right) \left\{ \left(E_n{}^m\right)^{n+1/2} \right\} = \left(\frac{[Q1]}{\Delta t} - W_{V2}[Q2^n] - W_2[Q3^n] \right) * \\ &\left\{ \left(E_n{}^m\right)^n \right\} + W_1\{RLS^{n+1}\} + W_2\{RLS^n\} + W_1\{B^{n+1}\} + W_2\{B^n\} \end{aligned} \qquad \textbf{(3.7.1.2.14)}$$

The boundary term {B} is calculated according to the specified boundary conditions the same as that in section 3.7.1.1.

3.7.1.3 Operator-Splitting Approach

According to the operator-splitting approach, equation (3.7.1.1.2) can be separated into two equations as follows.

$$\theta \frac{\left(E_n{}^m\right)^{n+1/2} - \left(E_n{}^m\right)^n}{\Delta t} + \nabla \cdot (\mathbf{V} E_n{}^m) - \nabla \cdot (\theta \mathbf{D} \cdot \nabla E_n{}^m) + \left(L_{HS} + \frac{\partial \theta}{\partial t} \right) E_n{}^m = R_{HS} \qquad \textbf{(3.7.1.3.1)}$$

$$\frac{E_n{}^{n+1} - [(E_n{}^m)^{n+1/2} + (E_n{}^{im})^n]}{\Delta t} = \theta R_{E_n}{}^{n+1} - \frac{\partial \ell n \theta}{\partial t} (E_n^{im})^{n+1} \qquad \textbf{(3.7.1.3.2)}$$

First, solve equation (3.7.1.3.1) and get $(E_n{}^m)^{n+1/2}$. Second, solve equation (3.7.1.3.2) together with algebraic equations representing equilibrium reactions using BIOGEOCHM scheme to obtain the individual species concentration.

Equation (3.7.1.3.1) can be solved through the same procedure as that in section 4.1.2, except for the load vectors {RLS}, which is calculated by the following equation.

$$RLS_i = \int_R N_i R_{HS} dR \qquad \textbf{(3.7.1.3.3)}$$

3.7.2 Application of the Finite Element Method to the Advective Form of the Reactive

Transport Equations

3.7.2.1 Fully-Implicit Scheme

Conversion of equation (2.7.22) to advection form is expressed as

$$\theta\frac{\partial E_n}{\partial t}+\frac{\partial\theta}{\partial t}E_n+\mathbf{V}\cdot\nabla E_n{}^m-\nabla\cdot(\theta\mathbf{D}\cdot\nabla E_n{}^m)+(\nabla\cdot\mathbf{V})E_n{}^m=M_{E_n^{as}}+\theta R_{E_n},\quad n\in[1,M\text{ - }N_E] \qquad \textbf{(3.7.2.1.1)}$$

According to equation (2.3.1), the right-hand side term R_{HS} and left hand side term L_{HS} can be assigned as follows.

$$\begin{aligned}&\mathit{If}\ \ q\le 0,\ \ M_{E_n^{as}}=qE_n{}^m,\quad L_{HS}=\left(-\mathbf{V}\bullet\ell n\left(\frac{\rho}{\rho_o}\right)-F\frac{\partial h}{\partial t}\right),\quad RHS=0\\&\mathit{Else}\ q>0,\ \ M_{E_n^{as}}=M_{E_n^{as}},\ L_{HS}=\left(q-\mathbf{V}\bullet\ell n\left(\frac{\rho}{\rho_o}\right)-F\frac{\partial h}{\partial t}\right),\quad R_{HS}=M_{E_n^{as}}\end{aligned} \qquad \textbf{(3.7.2.1.2)}$$

Then equation (3.7.2.1.1) is modified as

$$\theta\frac{\partial E_n}{\partial t}+\frac{\partial\theta}{\partial t}E_n+\mathbf{V}\cdot\nabla E_n{}^m-\nabla\cdot(\theta\mathbf{D}\cdot\nabla E_n{}^m)+L_{HS}E_n{}^m=R_{HS}+\theta R_{E_n} \qquad \textbf{(3.7.2.1.3)}$$

According to the fully-implicit scheme, equation (3.7.2.1.3) can be separated into two equations as follows.

$$\theta\frac{E_n{}^{n+1/2}-E_n{}^n}{\Delta t}+\frac{\partial\theta}{\partial t}E_n+\mathbf{V}\cdot\nabla E_n{}^m-\nabla\cdot(\theta\mathbf{D}\cdot\nabla E_n{}^m)+L_{HS}*E_n{}^m=R_{HS}+\theta R_{E_n} \qquad \textbf{(3.7.2.1.4)}$$

$$\frac{E_n{}^{n+1}-E_n{}^{n+1/2}}{\Delta t}=0 \qquad \textbf{(3.7.2.1.5)}$$

First, solve equation (3.7.2.1.4) and get $(E_n)^{n+1/2}$. Second, solve equation (3.7.2.1.5) together with algebraic equations representing equilibrium reactions using BIOGEOCHEM scheme to obtain the individual species concentration. Iteration is needed because reaction term in equation (3.7.2.1.4) needs to be updated by the results of (3.7.2.1.5).

Option 1: Express $E_n{}^m$ in terms of $(E_n{}^m/E_n)\ E_n$

$$\begin{aligned}&\theta\frac{\partial E_n}{\partial t}+\mathbf{V}\cdot\nabla\left(\frac{E_n{}^m}{E_n}E_n\right)-\nabla\cdot\left(\theta\mathbf{D}\cdot\frac{E_n{}^m}{E_n}\nabla E_n\right)\\&-\nabla\cdot\left[\theta\mathbf{D}\cdot\left(\nabla\frac{E_n{}^m}{E_n}\right)E_n\right]+\left(L_{HS}\frac{E_n{}^m}{E_n}+\frac{\partial\theta}{\partial t}\right)E_n=R_{HS}+\theta R_{E_n}\end{aligned} \qquad \textbf{(3.7.2.1.6)}$$

Use Galerkin or Petrov-Galerkin Finite-Element Method for the spatial descretization of transport equation. Integrate equation (3.7.2.1.6) in the spatial dimensions over the entire region as follows.

$$\int_R N_i \left[\theta \frac{\partial E_n}{\partial t} - \nabla \cdot \left(\theta \mathbf{D} \cdot \frac{E_n{}^m}{E_n} \nabla E_n \right) + \left(L_{HS} \frac{E_n{}^m}{E_n} + \frac{\partial \theta}{\partial t} \right) E_n \right] dR$$
$$+ \int_R W_i \left\{ \mathbf{V} \cdot \nabla \left(\frac{E_n{}^m}{E_n} E_n \right) - \nabla \cdot \left[\theta \mathbf{D} \cdot \left(\nabla \frac{E_n{}^m}{E_n} \right) E_n \right] \right\} dR = \int_R N_i (R_{HS} + \theta R_{E_n}) dR \quad \textbf{(3.7.2.1.7)}$$

Further, we obtain

$$\int_R N_i \theta \frac{\partial E_n}{\partial t} dR + \int_R W_\mathbf{i} \mathbf{V} \cdot \nabla \left(\frac{E_n{}^m}{E_n} E_n \right) dR + \int_R \nabla N_i \cdot \left(\theta \mathbf{D} \cdot \frac{E_n{}^m}{E_n} \nabla E_n \right) dR$$
$$+ \int_R \nabla W_i \cdot \left[\theta \mathbf{D} \cdot \left(\nabla \frac{E_n{}^m}{E_n} \right) E_n \right] dR + \int_R N_i \left(L_{HS} \frac{E_n{}^m}{E_n} + \frac{\partial \theta}{\partial t} \right) E_n dR = \int_R N_i (R_{HS} + \theta R_{E_n}) dR \quad \textbf{(3.7.2.1.8)}$$
$$+ \int_B \mathbf{n} \cdot \left(N_i \theta \mathbf{D} \cdot \frac{E_n{}^m}{E_n} \nabla E_n \right) dB + \int_B \mathbf{n} \cdot \left[W_i \theta \mathbf{D} \cdot \left(\nabla \frac{E_n{}^m}{E_n} \right) E_n \right] dB$$

Approximate solution E_n by a linear combination of the base functions as follows.

$$E_n \approx \hat{E}_n = \sum_{j=1}^{N} E_{nj}(t) N_j(R) \quad \textbf{(3.7.2.1.9)}$$

Substituting equation (3.7.2.1.9) into equation (3.7.2.1.8), we obtain

$$\sum_{j=1}^{N} \left[\left(\int_R N_i \theta N_j \right) \frac{\partial E_{nj}(t)}{\partial t} dR \right] + \sum_{j=1}^{N} \left\{ \left[\int_R W_i \mathbf{V} \cdot \left(\nabla \frac{E_n{}^m}{E_n} \right) N_j dR \right] E_{nj}(t) \right\}$$
$$+ \sum_{j=1}^{N} \left[\left(\int_R W_i \mathbf{V} \cdot \frac{E_n{}^m}{E_n} \nabla N_j dR \right) E_{nj}(t) \right] + \sum_{j=1}^{N} \left\{ \left\{ \int_R \nabla W_i \cdot \left[\theta \mathbf{D} \cdot \left(\nabla \frac{E_n{}^m}{E_n} \right) N_j \right] dR \right\} E_{nj}(t) \right\}$$
$$+ \sum_{j=1}^{N} \left\{ \left[\int_R \nabla N_i \cdot \left(\theta \mathbf{D} \cdot \frac{E_n{}^m}{E_n} \nabla N_j \right) dR + \int_R N_i \left(L_{HS} \frac{E_n{}^m}{E_n} + \frac{\partial \theta}{\partial t} \right) N_j dR \right] E_{nj}(t) \right\} \quad \textbf{(3.7.2.1.10)}$$
$$= \int_R N_i (R_{HS} + \theta R_{E_n}) dR + \int_B \mathbf{n} \cdot \left(N_i \theta \mathbf{D} \cdot \nabla E_n{}^m \right) dB$$

Equation (3.7.2.1.10) can be written in matrix form as

$$[Q1] \left\{ \frac{\partial E_n}{\partial t} \right\} + [Q2]\{E_n\} + [Q3]\{E_n\} = \{RLS\} + \{B\} \quad \textbf{(3.7.2.1.11)}$$

where the matrices [Q1], [Q2], [Q3] and load vectors {SS}, and {B} are given by

$$Q1_{ij} = \int_R N_i \theta^n N_j dR \quad \textbf{(3.7.2.1.12)}$$

$$Q2_{ij} = \int_R W_i \mathbf{V} \cdot \left(\nabla \frac{E_n{}^m}{E_n} \right) N_j dR$$
$$+ \int_R W_i \mathbf{V} \cdot \frac{E_n{}^m}{E_n} \nabla N_j dR + \int_R \nabla W_i \cdot \left[\theta \mathbf{D} \cdot \left(\nabla \frac{E_n{}^m}{E_n} \right) N_j \right] dR \quad \textbf{(3.7.2.1.13)}$$

$$Q3_{ij} = \int_R \nabla N_i \cdot \left(\theta \mathbf{D} \cdot \frac{E_n{}^m}{E_n} \nabla N_j \right) dR + \int_R N_i \left(L_{HS} \frac{E_n{}^m}{E_n} + \frac{\partial \theta}{\partial t} \right) N_j dR \quad \textbf{(3.7.2.1.14)}$$

$$RLS_i = \int_R N_i (R_{HS} + \theta R_{E_n}) dR \quad \textbf{(3.7.2.1.15)}$$

$$B_i = \int_B \mathbf{n} \cdot \left(N_i \theta \mathbf{D} \cdot \nabla E_n{}^m \right) dB \quad \textbf{(3.7.2.1.16)}$$

At n+1-th time step, equation (3.7.2.1.11) is approximated as

$$[Q1] \frac{\left\{ E_n{}^{n+1/2} \right\} - \left\{ E_n{}^n \right\}}{\Delta t} + W_{V1} [Q2^{n+1}] \left\{ E_n{}^{n+1/2} \right\} + W_{V2} [Q2^n] \left\{ E_n{}^n \right\} + W_1 [Q3^{n+1}] \left\{ E_n{}^{n+1/2} \right\}$$
$$+ W_2 [Q3^n] \left\{ E_n{}^n \right\} = W_1 \{RLS^{n+1}\} + W_2 \{RLS^n\} + W_1 \{B^{n+1}\} + W_2 \{B^n\} \quad \textbf{(3.7.2.1.17)}$$

So that

$$\left(\frac{[Q1]}{\Delta t} + W_{V1} [Q2^{n+1}] + W_1 [Q3^{n+1}] \right) \left\{ E_n{}^{n+1/2} \right\} = \left(\frac{[Q1]}{\Delta t} - W_{V2} [Q2^n] - W_2 [Q3^n] \right) \left\{ E_n{}^n \right\}$$
$$+ W_1 \{RLS^{n+1}\} + W_2 \{RLS^n\} + W_1 \{B^{n+1}\} + W_2 \{B^n\} \quad \textbf{(3.7.2.1.18)}$$

Option 2: Express $\mathbf{E_n^m}$ in terms of $\mathbf{E_n}$-$\mathbf{E_n^{im}}$

Use Galerkin or Petrov-Galerkin Finite-Element Method for the spatial descretization of transport equation. Integrate equation (3.7.2.1.6) in the spatial dimensions over the entire region as follows.

$$\int_R N_i \left[\theta \frac{\partial E_n}{\partial t} + \frac{\partial \theta}{\partial t} E_n - \nabla \cdot \left(\theta \mathbf{D} \cdot \nabla E_n{}^m \right) + L_{HS} \cdot E_n{}^m \right] dR + \int_R W_i \mathbf{V} \cdot \nabla E_n{}^m dR =$$
$$\int_R N_i (R_{HS} + \theta R_{E_n}) dR \quad \textbf{(3.7.2.1.19)}$$

Further, we obtain

$$\int_R N_i \theta \frac{\partial E_n}{\partial t} dR + \int_R N_i \frac{\partial \theta}{\partial t} E_n dR + \int_R W_i \mathbf{V} \cdot \nabla E_n{}^m dR + \int_R \nabla N_i \cdot \left(\theta \mathbf{D} \cdot \nabla E_n{}^m \right) dR +$$
$$\int_R N_i L_{HS} \cdot E_n{}^m dR = \int_R N_i (R_{HS} + \theta R_{E_n}) dR + \int_B \mathbf{n} \cdot \left(N_i \theta \mathbf{D} \cdot \nabla E_n{}^m \right) dB \quad \textbf{(3.7.2.1.20)}$$

Approximate solution E_n by a linear combination of the base functions as equation (3.7.2.1.9). Substituting equation (3.7.2.1.9) into equation (3.7.2.1.20), we obtain

$$\sum_{j=1}^{N}\left[\left(\int_R N_i\theta N_j\right)\frac{\partial E_{nj}(t)}{\partial t}dR\right]+\sum_{j=1}^{N}\left[\left(\int_R N_i\frac{\partial\theta}{\partial t}N_j\right)E_{nj}(t)dR\right]+$$
$$\sum_{j=1}^{N}\left\{\left[\int_R W_i\mathbf{V}\cdot\nabla N_j dR\right]E_{nj}{}^{m}(t)\right\}+\sum_{j=1}^{N}\left\{\left[\int_R\nabla N_i\cdot\left(\theta\mathbf{D}\cdot\nabla N_j\right)dR+\int_R N_iL_{HS}N_jdR\right]E_{nj}{}^{m}(t)\right\} \quad \textbf{(3.7.2.1.21)}$$
$$=\int_R N_i(R_{HS}+\theta R_{E_n})dR+\int_B \mathbf{n}\cdot\left(N_i\theta\mathbf{D}\cdot\nabla E_n{}^{m}\right)dB$$

Equation (3.7.2.1.21) can be written in matrix form as

$$[Q1]\left\{\frac{\partial E_n}{\partial t}\right\}+[Q4]\{E_n\}+[Q2]\left\{E_n{}^{m}\right\}+[Q3]\left\{E_n{}^{m}\right\}=\{RLS\}+\{B\} \quad \textbf{(3.7.2.1.22)}$$

where the matrices [Q1], [Q2], [Q3] and load vectors {SS}, and {B} are given by

$$Q1_{ij}=\int_R N_i\theta N_jdR, \qquad Q4_{ij}=\int_R N_i\frac{\partial\theta}{\partial t}N_jdR \quad \textbf{(3.7.2.1.23)}$$

$$Q2_{ij}=\int_R W_i\mathbf{V}\cdot\nabla N_jdR \quad \textbf{(3.7.2.1.24)}$$

$$Q3_{ij}=\int_R\nabla N_i\cdot\left(\theta\mathbf{D}\cdot\nabla N_j\right)dR+\int_R N_iL_{HS}N_jdR \quad \textbf{(3.7.2.1.25)}$$

$$RLS_i=\int_R N_i(R_{HS}+\theta R_{E_n})dR \quad \textbf{(3.7.2.1.26)}$$

$$B_i=\int_B\mathbf{n}\cdot\left(N_i\theta\mathbf{D}\cdot\nabla E_n{}^{m}\right)dB \quad \textbf{(3.7.2.1.27)}$$

Express $E_n{}^{m}$ in terms of E_n-$E_n{}^{im}$, equation (3.7.2.1.22) is modified as

$$[Q1]\left\{\frac{\partial E_n}{\partial t}\right\}+[Q4]\{E_n\}+[Q2]\{E_n\}+[Q3]\{E_n\}=$$
$$[Q2]\left\{E_n{}^{im}\right\}+[Q3]\left\{E_n{}^{im}\right\}+\{RLS\}+\{B\} \quad \textbf{(3.7.2.1.28)}$$

At n+1-th time step, equation (3.7.2.1.28) is approximated as

$$[Q1]\frac{\left\{E_n^{\ n+1/2}\right\}-\left\{E_n^{\ n}\right\}}{\Delta t}+[Q4]\left\{E_n^{\ n+1/2}\right\}+W_{V1}[Q2^{n+1}]\left\{E_n^{\ n+1/2}\right\}+W_{V2}[Q2^n]\left\{E_n^{\ n}\right\}$$
$$+W_1[Q3^{n+1}]\left\{E_n^{\ n+1/2}\right\}+W_2[Q3^n]\left\{E_n^{\ n}\right\}=W_{V1}[Q2^{n+1}]\left\{\left(E_n^{\ im}\right)^{n+1/2}\right\}$$
$$+W_{V2}[Q2^n]\left\{\left(E_n^{\ im}\right)^{n}\right\}+W_1[Q3^{n+1}]\left\{\left(E_n^{\ im}\right)^{n+1/2}\right\}+W_2[Q3^n]\left\{\left(E_n^{\ im}\right)^{n}\right\}$$
$$+W_1\{RLS^{n+1}\}+W_2\{RLS^n\}+W_1\{B^{n+1}\}+W_2\{B^n\} \quad \textbf{(3.7.2.1.29)}$$

So that

$$\left(\frac{[Q1]}{\Delta t}+[Q4]+W_{V1}[Q2^{n+1}]+W_1[Q3^{n+1}]\right)\left\{E_n^{\ n+1/2}\right\}=\frac{[Q1]}{\Delta t}\left\{E_n^{\ n}\right\}$$
$$-\left(W_{V2}[Q2^n]+W_2[Q3^n]\right)\left\{\left(E_n^{\ m}\right)^n\right\}+\left(W_{V1}[Q2^n]+W_1[Q3^n]\right)\left\{\left(E_n^{\ im}\right)^{n+1/2}\right\}+$$
$$W_1\{RLS^{n+1}\}+W_2\{RLS^n\}+W_1\{B^{n+1}\}+W_2\{B^n\} \quad \textbf{(3.7.2.1.30)}$$

For interior nodes i, B_i is zero, for boundary nodes i = b, B_i is calculated according to the specified boundary condition and shown as follows.

<u>Dirichlet boundary condition</u>

$$E_n^{\ m}=E_n^{\ m}(x_b,y_b,z_b,t) \quad \textbf{(3.7.2.1.31)}$$

<u>Variable boundary condition</u>

< Case 1 > when flow is going in from outside ($\mathbf{n}\cdot\mathbf{V}<0$)

$$\mathbf{n}\cdot(\mathbf{V}E_n^{\ m}-\theta\mathbf{D}\cdot\nabla E_n^{\ m})=\mathbf{n}\cdot\mathbf{V}E_n^{\ m}(x_b,y_b,z_b,t)$$
$$\Rightarrow\quad B_i=\int_{\mathbf{B}}\mathbf{n}\cdot N_i\mathbf{V}E_n^{\ m}dB-\int_{\mathbf{B}}\mathbf{n}\cdot N_i\mathbf{V}E_n^{\ m}(x_b,y_b,z_b,t)dB \quad \textbf{(3.7.2.1.32)}$$

< Case 2 > Flow is going out from inside ($\mathbf{n}\cdot\mathbf{V}>0$):

$$-\mathbf{n}\cdot(\theta\mathbf{D}\cdot\nabla E_n^{\ m})=0\quad\Rightarrow\quad B_i=0 \quad \textbf{(3.7.2.1.33)}$$

<u>Cauchy boundary condition</u>

$$n\cdot(\mathbf{V}E_n^{\ m}-\theta\mathbf{D}\cdot\nabla E_n^{\ m})=Q_{E_n^{\ m}}(x_b,y_b,z_b,t)$$
$$\Rightarrow\quad B_i=\int_{\mathbf{B}}\mathbf{n}\cdot N_i\mathbf{V}E_n^{\ m}dB-\int_{B}N_iQ_{E_n^{\ m}}(x_b,y_b,z_b,t)dB \quad \textbf{(3.7.2.1.34)}$$

<u>Neumann boundary condition</u>

$$-\mathbf{n}\cdot(\theta\mathbf{D}\cdot\nabla E_n^{\ m})=Q_{E_n^{\ m}}(x_b,y_b,z_b,t)\quad\Rightarrow\quad B_i=-\int_{B}N_iQ_{E_n^{\ m}}(x_b,y_b,z_b,t)dB \quad \textbf{(3.7.2.1.35)}$$

River/stream-subsurface interface boundary condition

$$\begin{aligned}&\mathbf{n}\cdot(\mathbf{V}E_n{}^m-\theta\mathbf{D}\cdot\nabla E_n{}^m)=\frac{\mathbf{n}\cdot\mathbf{V}}{2}\left\{[1+sign(\mathbf{n}\cdot\mathbf{V})]E_n{}^m+[1-sign(\mathbf{n}\cdot\mathbf{V})]\left(E_n{}^m\right)^{1D}\right\}\Rightarrow\\&B_i=\int_B\mathbf{n}\cdot N_i\mathbf{V}E_n{}^m dB-\int_B N_i\frac{\mathbf{n}\cdot\mathbf{V}}{2}\left\{[1+sign(\mathbf{n}\cdot\mathbf{V})]E_n{}^m+[1-sign(\mathbf{n}\cdot\mathbf{V})]\left(E_n{}^m\right)^{1D}\right\}dB\end{aligned}\qquad\textbf{(3.7.2.1.36)}$$

Overland-subsurface interface boundary condition

$$\begin{aligned}&\mathbf{n}\cdot(\mathbf{V}E_n{}^m-\theta\mathbf{D}\cdot\nabla E_n{}^m)=\frac{\mathbf{n}\cdot\mathbf{V}}{2}\left\{[1+sign(\mathbf{n}\cdot\mathbf{V})]E_n{}^m+[1-sign(\mathbf{n}\times\mathbf{V})]\left(E_n{}^m\right)^{2D}\right\}\Rightarrow\\&B_i=\int_{\mathbf{B}}\mathbf{n}\cdot N_i\mathbf{V}E_n{}^m dB-\int_B N_i\frac{\mathbf{n}\cdot\mathbf{V}}{2}\left\{[1+sign(\mathbf{n}\cdot\mathbf{V})]E_n{}^m+[1-sign(\mathbf{n}\cdot\mathbf{V})]\left(E_n{}^m\right)^{2D}\right\}dB\end{aligned}\qquad\textbf{(3.7.2.1.37)}$$

3.7.2.2 Mixed Predictor-Corrector and Operator-Splitting Method

According to the mixed predictor-corrector (on reaction rates) and operator-splitting (on immobile part of the kinetic variable) method, equation (3.7.2.1.3) can be separated into two equations as follows.

$$\begin{aligned}&\theta\frac{\left(E_n{}^m\right)^{n+1/2}-\left(E_n{}^m\right)^n}{\Delta t}+\frac{\partial\theta}{\partial t}E_n^m+\mathbf{V}\cdot\nabla E_n{}^m-\nabla\cdot(\theta\mathbf{D}\cdot\nabla E_n{}^m)+L_{HS}E_n{}^m=\\&R_{HS}+\theta R_{E_n}{}^n-\frac{\partial\theta}{\partial t}(E_n^{im})^n\end{aligned}\qquad\textbf{(3.7.2.2.1)}$$

$$\frac{E_n{}^{n+1}-[(E_n{}^m)^{n+1/2}+(E_n{}^{im})^n]}{\Delta t}=R_{E_n}{}^{n+1}-R_{E_n}{}^n-\frac{\partial\ell n\theta}{\partial t}(E_n^{im})^{n+1}+\frac{\partial\ell n\theta}{\partial t}(E_n^{im})^n\qquad\textbf{(3.7.2.2.2)}$$

First, solve equation (3.7.2.2.1) and get $(E_n{}^m)^{n+1/2}$. Second, solve equation (3.7.2.2.2) together with algebraic equations representing equilibrium reactions using BIOGEOCHM scheme to obtain the individual species concentration.

Use Galerkin or Petrov-Galerkin Finite-Element Method for the spatial descretization of transport equation. Integrate equation (3.7.2.2.1) in the spatial dimensions over the entire region as follows.

$$\begin{aligned}&\int_R N_i\left[\theta\frac{\partial E_n{}^m}{\partial t}-\nabla\cdot(\theta\mathbf{D}\cdot\nabla E_n{}^m)+\left(L_{HS}+\frac{\partial\theta}{\partial t}\right)E_n{}^m\right]dR+\int_R W_i\mathbf{V}\cdot\nabla E_n{}^m dR=\\&\int_R N_i\left(R_{HS}+\theta R_{E_n}{}^n-\frac{\partial\theta}{\partial t}(E_n^{im})^n\right)dR\end{aligned}\qquad\textbf{(3.7.2.2.3)}$$

Further, we obtain

$$\int_R N_i\theta\frac{\partial E_n{}^m}{\partial t}dR+\int_R W_i\mathbf{V}\cdot\nabla E_n{}^m dR+\int_R \nabla N_i\cdot(\theta\mathbf{D}\cdot\nabla E_n{}^m)dR+\int_R N_i\left(L_{HS}+\frac{\partial\theta}{\partial t}\right)E_n{}^m dR$$
$$=\int_R N_i\left(R_{HS}+\theta R_{E_n}{}^n-\frac{\partial\theta}{\partial t}(E_n^{im})^n\right)dR+\int_B \mathbf{n}\cdot N_i(\theta\mathbf{D}\cdot\nabla E_n{}^m)dB \quad \textbf{(3.7.2.2.4)}$$

Approximate solution E_n^m by a linear combination of the base functions as follows.

$$E_n{}^m\approx\hat{E}_n{}^m=\sum_{j=1}^{N}E_{nj}{}^m(t)N_j(R) \quad \textbf{(3.7.2.2.5)}$$

Substituting equation (3.7.2.2.5) into equation (3.7.2.2.4), we obtain

$$\sum_{j=1}^{N}\left[\left(\int_R N_i\theta^n N_j dR\right)\frac{\partial E_{nj}{}^m(t)}{\partial t}\right]+\sum_{j=1}^{N}\left[\left(\int_R W_i\mathbf{V}\cdot\nabla N_j dR\right)E_{nj}{}^m(t)\right]$$
$$+\sum_{j=1}^{N}\left\{\left[\int_R \nabla N_i\cdot(\theta\mathbf{D}\cdot\nabla N_j)dR+\int_R N_i\left(L_{HS}+\frac{\partial\theta}{\partial t}\right)N_j dR\right]E_{nj}{}^m(t)\right\} \quad \textbf{(3.7.2.2.6)}$$
$$=\int_R N_i\left(R_{HS}+\theta R_{E_n}{}^n-\frac{\partial\theta}{\partial t}(E_n^{im})^n\right)dR+\int_B \mathbf{n}\cdot N_i(\theta\mathbf{D}\cdot\nabla E_n{}^m)dB$$

Equation (3.7.2.2.6) can be written in matrix form as

$$[Q1]\left\{\frac{dE_n{}^m}{dt}\right\}+[Q2]\left\{E_n{}^m\right\}+[Q3]\left\{E_n{}^m\right\}=\{RLS\}+\{B\} \quad \textbf{(3.7.2.2.7)}$$

where the matrices [Q1], [Q2], and [Q3], and load vectors {RLS} and {B} are given by

$$Q1_{ij}=\int_R N_i\theta N_j dR \quad \textbf{(3.7.2.2.8)}$$

$$Q2_{ij}=\int_R W_i\mathbf{V}\cdot\nabla N_j dR \quad \textbf{(3.7.2.2.9)}$$

$$Q3_{ij}=\int_R \nabla N_i\cdot\left(\theta\mathbf{D}\cdot\nabla N_j\right)dR+\int_R N_i\left(L_{HS}+\frac{\partial\theta}{\partial t}\right)N_j dR \quad \textbf{(3.7.2.2.10)}$$

$$RLS_i=\int_R N_i\left(R_{HS}+\theta R_{E_n}{}^n-\frac{\partial\theta}{\partial t}(E_n^{im})^n\right)dR \quad \textbf{(3.7.2.2.11)}$$

$$B_i=\int_B \mathbf{n}\cdot(N_i\theta\mathbf{D}\cdot\nabla E_n{}^m)dB \quad \textbf{(3.7.2.2.12)}$$

At n+1-th time step, equation (3.7.2.2.7) is approximated as

$$
\begin{aligned}
&[Q1]\frac{\left\{ \left(E_n{}^m\right)^{n+1/2}\right\} - \left\{ \left(E_n{}^m\right)^{n}\right\}}{\Delta t} + W_{V1}[Q2^{n+1}]\left\{ \left(E_n{}^m\right)^{n+1/2}\right\} \\
&+W_{V2}[Q2^n]\left\{ \left(E_n{}^m\right)^{n}\right\} + W_1[Q3^{n+1}]\left\{ \left(E_n{}^m\right)^{n+1/2}\right\} + W_2[Q3^n]\left\{ \left(E_n{}^m\right)^{n}\right\} \\
&= W_1\{RLS^{n+1}\} + W_2\{RLS^n\} + W_1\{B^{n+1}\} + W_2\{B^n\}
\end{aligned}
\quad \textbf{(3.7.2.2.13)}
$$

So that

$$
\begin{aligned}
&\left(\frac{[Q1^n]}{\Delta t} + W_{V1}[Q2^{n+1}] + W_1[Q3^{n+1}]\right)\left\{ \left(E_n{}^m\right)^{n+1/2}\right\} = \left(\frac{[Q1^n]}{\Delta t} - W_{V2}[Q2^n] - W_2[Q3^n]\right) * \\
&\left\{ \left(E_n{}^m\right)^{n}\right\} + W_1\{RLS^{n+1}\} + W_2\{RLS^n\} + W_1\{B^{n+1}\} + W_2\{B^n\}
\end{aligned}
\quad \textbf{(3.7.2.2.14)}
$$

The boundary term {B} is calculated according the same as that in section 3.7.2.1.

3.7.3 Operator-Splitting Approach

According to the operator-splitting approach, equation (3.7.2.1.2) can be separated into two equations as follows.

$$
\theta\frac{\left(E_n{}^m\right)^{n+1/2} - \left(E_n{}^m\right)^{n}}{\Delta t} + \mathbf{V}\cdot\nabla E_n{}^m - \nabla\cdot(\theta\mathbf{D}\cdot\nabla E_n{}^m) + \left(L_{HS} + \frac{\partial\theta}{\partial t}\right)E_n{}^m = R_{HS} \quad \textbf{(3.7.2.3.1)}
$$

$$
\frac{E_n{}^{n+1} - [(E_n{}^m)^{n+1/2} + (E_n{}^{im})^n]}{\Delta t} = R_{E_n}{}^{n+1} - \frac{\partial \ell n\theta}{\partial t}(E_n^{im})^{n+1} \quad \textbf{(3.7.2.3.2)}
$$

First, solve equation (3.7.2.3.1) and get $(E_n{}^m)^{n+1/2}$. Second, solve equation (3.7.2.3.2) together with algebraic equations representing equilibrium reactions using BIOGEOCHM scheme to obtain the individual species concentration.

Equation (3.7.2.3.1) can be solved through the same procedure as that in section 4.1.2, except for the load vectors {RLS}, which is calculated by the following equation.

$$
RLS_i = \int_R N_i R_{HS} dR \quad \textbf{(3.7.2.3.3)}
$$

3.7.4 Application of the Modified Lagrangian-Eulerian Approach to the Largrangian Form of the Reactive Transport Equations

3.7.4.1 Fully-Implicit Scheme

Option 1: Express $E_n{}^m$ in terms of $(E_n{}^m/E_n)\,E_n$

Express $E_n{}^m$ in terms of $(E_n{}^m/E_n)\,E_n$ to make E_n's as primary dependent variables, equation

(3.7.2.1.4) is modified as

$$\theta\frac{\partial E_n}{\partial t}+\frac{\partial \theta}{\partial t}E_n+\left[\mathbf{V}\frac{E_n{}^m}{E_n}-\theta\mathbf{D}\cdot\left(\nabla\frac{E_n{}^m}{E_n}\right)\right]\cdot\nabla(E_n)-\nabla\cdot\left(\theta\mathbf{D}\cdot\frac{E_n{}^m}{E_n}\nabla E_n\right)$$
$$\left\{\mathbf{V}\cdot\nabla\left(\frac{E_n{}^m}{E_n}\right)-\nabla\cdot\left[\theta\mathbf{D}\cdot\left(\nabla\frac{E_n{}^m}{E_n}\right)\right]+L_{HS}\frac{E_n{}^m}{E_n}\right\}E_n=R_{HS}+\theta R_{E_n} \tag{3.7.3.1.1}$$

Assign the particle tracking velocity $\mathbf{V}_{\text{track}}$ as follows

$$\mathbf{V}_{track}=\frac{1}{\theta}\left[\mathbf{V}\frac{E_n{}^m}{E_n}-\theta\mathbf{D}\cdot\left(\nabla\frac{E_n{}^m}{E_n}\right)\right] \tag{3.7.3.1.2}$$

Equation (3.7.3.1.1) in Lagrangian-Eulerian form is written as

In Lagrangian step,

$$\frac{DE_n}{D\tau}=\frac{\partial E_n}{\partial t}+\mathbf{V}_{track}\cdot\nabla E_n=0 \tag{3.7.3.1.3}$$

In Eulerian step,

$$\frac{DE_n}{D\tau}-D+KE_n=R_L \tag{3.7.3.1.4}$$

where

$$\theta D=\nabla\cdot\left(\theta\mathbf{D}\frac{E_n{}^m}{E_n}\cdot\nabla E_n\right) \tag{3.7.3.1.5}$$

$$K=\frac{1}{\theta}\left\{\mathbf{V}\cdot\nabla\left(\frac{E_n{}^m}{E_n}\right)-\nabla\cdot\left[\theta\mathbf{D}\cdot\left(\nabla\frac{E_n{}^m}{E_n}\right)\right]+\left(\frac{\partial\theta}{\partial t}+L_{HS}\frac{E_n{}^m}{E_n}\right)\right\} \tag{3.7.3.1.6}$$

$$R_L=\frac{1}{\theta}\left(R_{HS}+\theta R_{E_n}\right) \tag{3.7.3.1.7}$$

The integration of equation (3.7.3.1.5) can be written as

$$\int_R N_i\theta DdR=-\int_R\nabla N_i\cdot(\theta\mathbf{D}\frac{E_n{}^m}{E_n}\cdot\nabla E_n)dR+\int_B\mathbf{n}\cdot N_i(\theta\mathbf{D}\frac{E_n{}^m}{E_n}\cdot\nabla E_n)dB \tag{3.7.3.1.8}$$

Approximate D and E_n by linear combination of the base functions as follows.

$$D\approx\hat{D}=\sum_{j=1}^{N}D_j(t)N_j(R) \tag{3.7.3.1.9}$$

$$E_n \approx \hat{E}_n = \sum_{j=1}^{N} E_{nj}(t) N_j(R) \tag{3.7.3.1.10}$$

Put Equations (3.7.3.1.9) and (3.7.3.1.10) into Equation (3.7.3.1.8), we obtain

$$\sum_{j=1}^{N}\left[\left(\int_R N_i \theta N_j dR\right) * D_j\right] = -\sum_{j=1}^{N}\left[\left(\int_R \nabla N_i \cdot (\theta \mathbf{D} \frac{E_n{}^m}{E_n} \cdot \nabla N_j) dR\right) E_{nj}\right] + \int_B \mathbf{n} \cdot N_i (\theta \mathbf{D} \frac{E_n{}^m}{E_n} \cdot \nabla E_n) dB \tag{3.7.3.1.11}$$

Assign matrices [QA] and [QD] and load vector {B} as following.

$$QA_{ij} = \int_R N_i \theta N_j dR \tag{3.7.3.1.12}$$

$$QD_{ij} = \int_R \nabla N_i \cdot (\theta \mathbf{D} \frac{E_n{}^m}{E_n} \cdot \nabla N_j) dR \tag{3.7.3.1.13}$$

$$B_i = \int_B \mathbf{n} \cdot N_i (\theta \mathbf{D} \frac{E_n{}^m}{E_n} \cdot \nabla E_n) dB \tag{3.7.3.1.14}$$

Equation (3.7.3.1.11) is expressed as

$$[QA]\{D\} = -[QD]\{E_n\} + \{B\} \tag{3.7.3.1.15}$$

Lump matrix [QA] into diagonal matrix and update

$$QD_{ij} = QD_{ij} \big/ QA_{ii} \tag{3.7.3.1.16}$$

$$B_i = \int_B \mathbf{n} \cdot N_i (\theta \mathbf{D} \cdot \nabla E_n{}^m) dB \Big/ QA_{ii} - \int_B \mathbf{n} \cdot N_i (\theta \mathbf{D} \cdot \nabla \frac{E_n{}^m}{E_n} E_n) dB \Big/ QA_{ii} \tag{3.7.3.1.17}$$

Then

$$\{D\} = -[QD]\{E_n\} + \{B\} \tag{3.7.3.1.18}$$

Equation (3.7.3.1.4) written in matrix form is then expressed as

$$\left(\frac{[U]}{\Delta \tau} + W_1[QD^{n+1}] + W_1\left[K^{n+1}\right]\right)\left\{E_n{}^{n+1/2}\right\} = \frac{[U]}{\Delta \tau}\left\{E_n{}^*\right\} - W_2\left(\left[K\right]\left\{E_n{}^m\right\}\right)^* + W_2\{D\}^* + W_1\left\{R_L{}^{n+1}\right\} + W_2\left\{R_L{}^*\right\} + W_1\left\{B^{n+1}\right\} \tag{3.7.3.1.19}$$

where [U] is the unit matrix, $\Delta\tau$ is the tracking time, W_1 and W_2 are time weighting factors, matrices and vectors with $^{n+1}$ and $^{n+1/2}$ are evaluated over the region at the new time step n+1. Matrices and vectors with superscript * corresponds to the n-th time step values interpolated at the location where a node is tracked through particle tracking in Lagrangian step.

For interior nodes i, B_i is zero, for boundary nodes i = b, B_i is calculated according to the specified boundary condition and shown as follows.

Dirichlet boundary condition

$$E_n{}^m = E_n{}^m(x_b, y_b, z_b, t) \;\Rightarrow$$
$$B_i = \int_B \mathbf{n}\cdot N_i(\theta\mathbf{D}\cdot\nabla E_n{}^m)dB \Big/ QA_{ii} - \int_B \mathbf{n}\cdot N_i(\theta\mathbf{D}\cdot\nabla \frac{E_n{}^m}{E_n} E_n)dB \Big/ QA_{ii} \quad \textbf{(3.7.3.1.20)}$$

Variable boundary condition

< Case 1 > when flow is going in from outside (**n·V** < 0)

$$\mathbf{n}\cdot(\mathbf{V}E_n{}^m - \theta\mathbf{D}\cdot\nabla E_n{}^m) = \mathbf{n}\cdot VE_n{}^m(x_b, y_b, z_b, t) \;\Rightarrow\; B_i = \int_{\mathbf{B}} \mathbf{n}\cdot N_i\mathbf{V}E_n{}^m dB \Big/ QA_{ii}$$
$$-\int_B \mathbf{n}\cdot N_i\mathbf{V}E_n{}^m(x_b, y_b, z_b, t)dB \Big/ QA_{ii} - \int_B \mathbf{n}\cdot N_i(\theta\mathbf{D}\cdot\nabla\frac{E_n{}^m}{E_n}E_n)dB \Big/ QA_{ii} \quad \textbf{(3.7.3.1.21)}$$

< Case 2 > Flow is going out from inside (**n·V** > 0):

$$-\mathbf{n}\cdot(\theta\mathbf{D}\cdot\nabla E_n{}^m) = 0 \;\Rightarrow\; B_i = -\int_B \mathbf{n}\cdot N_i(\theta\mathbf{D}\cdot\nabla\frac{E_n{}^m}{E_n}E_n)dB \Big/ QA_{ii} \quad \textbf{(3.7.3.1.22)}$$

Cauchy boundary condition

$$n\cdot(VE_n{}^m - \theta D\cdot\nabla E_n{}^m) = Q_{E_n{}^m}(x_b, y_b, z_b, t) \;\Rightarrow\; B_i = \int_B \mathbf{n}\cdot N_i\mathbf{V}E_n{}^m dB \Big/ QA_{ii}$$
$$-\int_B N_i Q_{E_n{}^m}(x_b, y_b, z_b, t)dB \Big/ QA_{ii} - \int_B \mathbf{n}\cdot N_i(\theta\mathbf{D}\cdot\nabla\frac{E_n{}^m}{E_n}E_n)dB \Big/ QA_{ii} \quad \textbf{(3.7.3.1.23)}$$

Neumann boundary condition

$$-\mathbf{n}\cdot(\theta\mathbf{D}\cdot\nabla E_n{}^m) = Q_{E_n{}^m}(x_b, y_b, z_b, t) \;\Rightarrow\; B_i = -\int_B N_i Q_{E_n{}^m}(x_b, y_b, z_b, t)dB \Big/ QA_{ii}$$
$$-\int_B \mathbf{n}\cdot N_i(\theta\mathbf{D}\cdot\nabla\frac{E_n{}^m}{E_n}E_n)dB \Big/ QA_{ii} \quad \textbf{(3.7.3.1.24)}$$

River/stream-subsurface interface boundary condition

$$\mathbf{n}\cdot(\mathbf{V}E_n{}^m - \theta\mathbf{D}\cdot\nabla E_n{}^m) = \frac{\mathbf{n}\cdot\mathbf{V}}{2}\left\{ [1+sign(\mathbf{n}\cdot\mathbf{V})]E_n{}^m + [1-sign(\mathbf{n}\cdot\mathbf{V})]\left(E_n{}^m\right)^{1D} \right\}$$

$$\Rightarrow \quad B_i = \int_B \mathbf{n}\cdot N_i \mathbf{V} E_n{}^m dB \Big/ QA_{ii} - \int_B \mathbf{n}\cdot N_i (\theta\mathbf{D}\cdot\nabla\frac{E_n{}^m}{E_n}E_n)dB \Big/ QA_{ii} \tag{3.7.3.1.25}$$

$$-\int_B N_i \frac{\mathbf{n}\cdot\mathbf{V}}{2}\left\{ [1+sign(\mathbf{n}\cdot\mathbf{V})]E_n{}^m + [1-sign(\mathbf{n}\cdot\mathbf{V})]\left(E_n{}^m\right)^{1D} \right\} dB \Big/ QA_{ii}$$

Overland-subsurface interface boundary condition

$$\mathbf{n}\cdot(\mathbf{V}E_n{}^m - \theta\mathbf{D}\cdot\nabla E_n{}^m) = \frac{\mathbf{n}\cdot\mathbf{V}}{2}\left\{ [1+sign(\mathbf{n}\cdot\mathbf{V})]E_n{}^m + [1-sign(\mathbf{n}\cdot\mathbf{V})]\left(E_n{}^m\right)^{2D} \right\}$$

$$\Rightarrow \quad B_i = \int_B \mathbf{n}\cdot N_i \mathbf{V} E_n{}^m dB \Big/ QA_{ii} - \int_B \mathbf{n}\cdot N_i (\theta\mathbf{D}\cdot\nabla\frac{E_n{}^m}{E_n}E_n)dB \Big/ QA_{ii} \tag{3.7.3.1.26}$$

$$-\int_B N_i \frac{\mathbf{n}\cdot\mathbf{V}}{2}\left\{ [1+sign(\mathbf{n}\cdot\mathbf{V})]E_n{}^m + [1-sign(\mathbf{n}\cdot\mathbf{V})]\left(E_n{}^m\right)^{2D} \right\} dB \Big/ QA_{ii}$$

Option 2: Express $E_n{}^m$ in terms of E_n-$E_n{}^m$

Express $E_n{}^m$ in terms of E_n-$E_n{}^m$ to make E_n's as primary dependent variables, equation (3.7.2.1.4) is modified as

$$\begin{aligned} &\theta\frac{\partial E_n}{\partial t} + \frac{\partial \theta}{\partial t}E_n + \mathbf{V}\cdot\nabla E_n - \nabla\cdot\left(\theta\mathbf{D}\cdot\nabla E_n\right) + L_{HS}E_n \\ &= \mathbf{V}\cdot\nabla E_n{}^{im} - \nabla\cdot\left(\theta\mathbf{D}\cdot\nabla E_n{}^{im}\right) + L_{HS}E_n{}^{im} + R_{HS} + \theta R_{E_n} \end{aligned} \tag{3.7.3.1.27}$$

Assign the particle tracking velocity $\mathbf{V}_{track}$ as follows

$$\mathbf{V}_{track} = \frac{1}{\theta}\mathbf{V} \tag{3.7.3.1.28}$$

Equation (3.7.3.1.27) in Lagrangian-Eulerian form is written as

In Lagrangian step,

$$\frac{DE_n}{D\tau} = \frac{\partial E_n}{\partial t} + \mathbf{V}_{track}\cdot\nabla E_n = 0 \tag{3.7.3.1.29}$$

In Eulerian step,

$$\frac{DE_n}{D\tau} - D + KE_n = T + R_L \tag{3.7.3.1.30}$$

where

$$\theta D = \nabla \cdot (\theta \mathbf{D} \cdot \nabla E_n) \qquad \textbf{(3.7.3.1.31)}$$

$$K = \frac{L_{HS} + \dfrac{\partial \theta}{\partial t}}{\theta} \qquad \textbf{(3.7.3.1.32)}$$

$$\theta T = \mathbf{V} \cdot \nabla E_n^{\ im} - \nabla \cdot \left(\theta \mathbf{D} \cdot \nabla E_n^{\ im} \right) \qquad \textbf{(3.7.3.1.33)}$$

$$R_L = \frac{1}{\theta^n} \left(L_{HS} E_n^{\ im} + R_{HS} + \theta R_{E_n} \right) \qquad \textbf{(3.7.3.1.34)}$$

The integration of equation (3.7.3.1.31) can be written as

$$\int_R N_i \theta D dR = -\int_R \nabla N_i \cdot (\theta \mathbf{D} \cdot \nabla E_n) dR + \int_B \mathbf{n} \cdot N_i (\theta \mathbf{D} \cdot \nabla E_n) dB \qquad \textbf{(3.7.3.1.35)}$$

Approximate D and E_n by linear combination of the base functions as follows.

$$D \approx \hat{D} = \sum_{j=1}^{N} D_j(t) N_j(R) \qquad \textbf{(3.7.3.1.36)}$$

$$E_n \approx \hat{E}_n = \sum_{j=1}^{N} E_{nj}(t) N_j(R) \qquad \textbf{(3.7.3.1.37)}$$

Put Equations (3.7.3.1.36) and (3.7.3.1.37) into Equation (3.7.3.1.35), we obtain

$$\begin{aligned} &\sum_{j=1}^{N} \left[\left(\int_R N_i \theta N_j dR \right) D_j \right] \\ &= -\sum_{j=1}^{N} \left[\left(\int_R \nabla N_i \cdot (\theta \mathbf{D} \cdot \nabla N_j) dR \right) E_{nj} \right] + \int_B \mathbf{n} \cdot N_i (\theta \mathbf{D} \cdot \nabla E_n) dB \end{aligned} \qquad \textbf{(3.7.3.1.38)}$$

Assign matrices [QA] and [QD] and load vector {B} as following.

$$QA_{ij} = \int_R N_i \theta N_j dR \qquad \textbf{(3.7.3.1.39)}$$

$$QD_{ij} = \int_R \nabla N_i \cdot (\theta \mathbf{D} \cdot \nabla N_j) dR \qquad \textbf{(3.7.3.1.40)}$$

$$B1_i = \int_B \mathbf{n} \cdot N_i (\theta \mathbf{D} \cdot \nabla E_n) dB \qquad \textbf{(3.7.3.1.41)}$$

Equation (3.7.3.1.31) is expressed as

$$[QA]\{D\} = -[QD]\{E_n\} + \{B1\} \tag{3.7.3.1.42}$$

Similarly,

$$[QA]\{T\} = [QT]\{E_n^{\ im}\} + \{B2\} \tag{3.7.3.1.43}$$

where

$$QT_{ij} = \int_R N_i \mathbf{V} \cdot \nabla N_j dR - \int_R \nabla N_i \cdot (\theta \mathbf{D} \cdot \nabla N_j) dR \tag{3.7.3.1.44}$$

$$B2_i = -\int_B \mathbf{n} \cdot N_i (\theta \mathbf{D} \cdot \nabla E_n^{\ im}) dB \tag{3.7.3.1.45}$$

Lump matrix [QA] into diagonal matrix and update

$$QD_{ij} = QD_{ij} / QA_{ii} \tag{3.7.3.1.46}$$

$$B1_i = B1_i / QA_{ii} \tag{3.7.3.1.47}$$

$$QT_{ij} = QT_{ij} / QA_{ii} \tag{3.7.3.1.48}$$

$$B2_i = B2_i / QA_{ii} \tag{3.7.3.1.49}$$

Then

$$\{D\} = -[QD]\{E_n\} + \{B1\} \tag{3.7.3.1.50}$$

$$\{T\} = [QT]\{E_n^{\ im}\} + \{B2\} \tag{3.7.3.1.51}$$

Assign

$$B_i = B1_i + B2_i = \int_B \mathbf{n} \cdot N_i (\theta \mathbf{D} \cdot \nabla E_n^{\ m}) dB \Big/ QA_{ii} \tag{3.7.3.1.52}$$

So that

$$\{D\} + \{T\} = -[QD]\{E_n\} + [QT]\{E_n^{\ im}\} + \{B\} \tag{3.7.3.1.53}$$

Equation (3.7.3.1.30) written in matrix form is then expressed as

$$\begin{aligned} &\left(\frac{[U]}{\Delta \tau} + W_1 [QD^{n+1}] + W_1 \left[K^{n+1} \right] \right) \left\{ E_n^{\ n+1/2} \right\} = \frac{[U]}{\Delta \tau} \left\{ E_n^{\ *} \right\} - W_2 \left([K] \{ E_n \} \right)^* \\ &+ W_1 [QT^{n+1}] \left\{ \left(E_n^{\ im} \right)^{n+1} \right\} + W_2 (\{D\} + \{T\})^* + W_1 \left\{ R_L^{\ n+1} \right\} + W_2 \left\{ R_L^{\ *} \right\} + W_1 \left\{ B^{n+1} \right\} \end{aligned} \tag{3.7.3.1.54}$$

For interior nodes i, B_i is zero, for boundary nodes i = b, B_i is calculated according to the specified boundary condition and shown as follows.

Dirichlet boundary condition

$$E_n{}^m = E_n{}^m(x_b, y_b, z_b, t) \;\;\Rightarrow\;\; B_i = \int_B \mathbf{n}\cdot N_i(\theta\mathbf{D}\cdot\nabla E_n{}^m)dB \Big/ QA_{ii} \tag{3.7.3.1.55}$$

Variable boundary condition

< Case 1 > when flow is going in from outside ($\mathbf{n}\cdot\mathbf{V} < 0$)

$$\begin{gathered}\mathbf{n}\cdot(\mathbf{V}E_n{}^m - \theta\mathbf{D}\cdot\nabla E_n{}^m) = \mathbf{n}\cdot\mathbf{V}E_n{}^m(x_b, y_b, z_b, t) \\ \Rightarrow\;\; B_i = \int_B \mathbf{n}\cdot N_i\mathbf{V}E_n{}^m dB \Big/ QA_{ii} - \int_{\mathbf{B}} \mathbf{n}\cdot N_i\mathbf{V}E_n{}^m(x_b, y_b, z_b, t)dB \Big/ QA_{ii}\end{gathered} \tag{3.7.3.1.56}$$

< Case 2 > Flow is going out from inside ($\mathbf{n}\cdot\mathbf{V} > 0$):

$$-n\cdot(\theta\mathbf{D}\cdot\nabla E_n{}^m) = 0 \;\;\Rightarrow\;\; B_i = 0 \tag{3.7.3.1.57}$$

Cauchy boundary condition

$$\begin{gathered}\mathbf{n}\cdot(\mathbf{V}E_n{}^m - \theta\mathbf{D}\cdot\nabla E_n{}^m) = Q_{E_n{}^m}(x_b, y_b, z_b, t) \\ \Rightarrow\;\; B_i = \int_B \mathbf{n}\cdot N_i\mathbf{V}E_n{}^m dB \Big/ QA_{ii} - \int_B N_i Q_{E_n{}^m}(x_b, y_b, z_b, t)dB \Big/ QA_{ii}\end{gathered} \tag{3.7.3.1.58}$$

Neumann boundary condition

$$-\mathbf{n}\cdot(\theta\mathbf{D}\cdot\nabla E_n{}^m) = Q_{E_n{}^m}(x_b, y_b, z_b, t) \;\;\Rightarrow\;\; B_i = -\int_B N_i Q_{E_n{}^m}(x_b, y_b, z_b, t)dB \Big/ QA_{ii} \tag{3.7.3.1.59}$$

River/stream-subsurface interface boundary condition

$$\begin{gathered}\mathbf{n}\cdot(\mathbf{V}E_n{}^m - \theta\mathbf{D}\cdot\nabla E_n{}^m) = \frac{\mathbf{n}\cdot\mathbf{V}}{2}\left\{ [1 + sign(\mathbf{n}\cdot\mathbf{V})]E_n{}^m + [1 - sign(\mathbf{n}\cdot\mathbf{V})]\left(E_n{}^m\right)^{1D}\right\} \\ \Rightarrow\;\; B_i = \int_B \mathbf{n}\cdot N_i\mathbf{V}E_n{}^m dB \Big/ QA_{ii} \\ -\int_B N_i\frac{\mathbf{n}\cdot\mathbf{V}}{2}\left\{ [1 + sign(\mathbf{n}\cdot\mathbf{V})]E_n{}^m + [1 - sign(\mathbf{n}\cdot\mathbf{V})]\left(E_n{}^m\right)^{1D}\right\}dB \Big/ QA_{ii}\end{gathered} \tag{3.7.3.1.60}$$

Overland-subsurface interface boundary condition

$$\mathbf{n}\cdot(\mathbf{V}E_n{}^m - \theta\mathbf{D}\cdot\nabla E_n{}^m) = \frac{\mathbf{n}\cdot\mathbf{V}}{2}\left\{ [1+sign(\mathbf{n}\cdot\mathbf{V})]E_n{}^m + [1-sign(\mathbf{n}\cdot\mathbf{V})]\left(E_n{}^m\right)^{2D}\right\}$$

$$\Rightarrow \quad B_i = \int_B \mathbf{n}\cdot N_i \mathbf{V} E_n{}^m dB \Big/ QA_{ii} \tag{3.7.3.1.61}$$

$$-\int_B N_i \frac{\mathbf{n}\cdot\mathbf{V}}{2}\left\{ [1+sign(\mathbf{n}\cdot\mathbf{V})]E_n{}^m + [1-sign(\mathbf{n}\cdot\mathbf{V})]\left(E_n{}^m\right)^{2D}\right\} dB \Big/ QA_{ii}$$

At upstream flux boundary nodes, equation (3.7.3.1.19) and (3.7.3.1.54) cannot be applied because $\Delta\tau$ equals zero. Thus, we propose a modified LE approach in which the matrix equation for upstream boundary nodes is obtained by explicitly applying the finite element method to the boundary conditions. For example, at the upstream variable boundary

$$\int_B N_i \mathbf{n}\cdot(\mathbf{V}E_n{}^m - \theta\mathbf{D}\cdot\nabla E_n{}^m) dB = \int_B N_i \mathbf{n}\cdot\mathbf{V} E_n{}^m (x_b, y_b, z_b, t) dB \tag{3.7.3.1.62}$$

So that the following matrix equation can be assembled at the boundary nodes

$$[QF]\{E_n{}^m\} = [QB]\{B\} \tag{3.7.3.1.63}$$

in which

$$QF_{ij} = \int_B (N_i \mathbf{n}\cdot\mathbf{V} N_j - N_i \mathbf{n}\cdot\theta\mathbf{D}\cdot\nabla N_j) dB \tag{3.7.3.1.64}$$

$$QB_{ij} = \int_B N_i \mathbf{n}\cdot\mathbf{V} N_j dB \tag{3.7.3.1.65}$$

$$B_j = E_n{}^m{}_j (x_b, y_b, z_b, t) \tag{3.7.3.1.66}$$

where $E_n{}^m{}_j(x_b, y_b, z_b, t)$ is the value of $E_n{}^m(x_b, y_b, z_b, t)$ evaluated at point j.

3.7.4.2 Mixed Predictor-Corrector and Operator-Splitting Method

Equation (3.7.2.2.1) in Lagrangian-Eulerian form is written as follows.

In Lagrangian step,

$$\frac{DE_n{}^m}{D\tau} = \frac{\partial E_n{}^m}{\partial t} + \mathbf{V}_{track}\cdot\nabla E_n{}^m = 0 \tag{3.7.3.2.1}$$

where particle tracking velocity is $\mathbf{V}_{\text{track}}$ is defined in Equation (3.7.3.1.28) .

In Eulerian step,

$$\frac{DE_n{}^m}{D\tau} - D + KE_n{}^m = R_L \tag{3.7.3.2.2}$$

where

$$\theta D = \nabla \cdot (\theta \mathbf{D} \cdot \nabla E_n{}^m) \tag{3.7.3.2.3}$$

$$K = \frac{L_{HS} + \frac{\partial \theta}{\partial t}}{\theta} \tag{3.7.3.2.4}$$

$$R_L = \frac{1}{\theta}\left(R_{HS} + \theta R_{E_n}{}^n - \frac{\partial \theta}{\partial t}(E_n^{im})^n \right) \tag{3.7.3.2.5}$$

According to equation (3.7.3.1.50)

$$[QA]\{D\} = -[QD]\{E_n{}^m\} + \{B\} \tag{3.7.3.2.6}$$

$$QA_{ij} = \int_R N_i \theta^n N_j dR \tag{3.7.3.2.7}$$

$$QD_{ij} = \int_R \nabla N_i \cdot (\theta \mathbf{D} \cdot \nabla N_j) dR \tag{3.7.3.2.8}$$

$$B_i = \int_B n \cdot N_i (\theta \mathbf{D} \cdot \nabla E_n{}^m) dB \tag{3.7.3.2.9}$$

Lump matrix [QA] into diagonal matrix and update

$$QD_{ij} = QD_{ij} / QA_{ii} \tag{3.7.3.2.10}$$

$$B_i = B_i / QA_{ii} \tag{3.7.3.2.11}$$

Then

$$\{D\} = -[QD]\{E_n{}^m\} + \{B\} \tag{3.7.3.2.12}$$

Equation (3.7.3.2.2) written in matrix form is then expressed as

$$\begin{aligned} &\left(\frac{[U]}{\Delta \tau} + W_1 [QD^{n+1}] + W_1 [K^{n+1}] \right) \left\{ \left(E_n{}^m \right)^{n+1/2} \right\} = \frac{[U]}{\Delta \tau} \left\{ \left(E_n{}^m \right)^* \right\} \\ &+ W_2 \left\{ D^* \right\} - W_2 \left([K] \left\{ E_n{}^m \right\} \right)^* + W_1 \left\{ RL^{n+1} \right\} + W_2 \left\{ RL^* \right\} + W_1 \left\{ B^{n+1} \right\} \end{aligned} \tag{3.7.3.2.13}$$

At upstream flux boundary nodes, equation (3.7.3.2.13) cannot be applied because $\Delta\tau$ equals zero.

Thus, we propose a modified LE approach in which the matrix equation for upstream boundary nodes is obtained by explicitly applying the finite element method to the boundary conditions as in Section 3.7.3.1.

3.7.4.3 Operator-Splitting Approach

Equation (3.7.2.3.1) can be solved through the same procedure as that in section 4.5.2, except that

$$RL = \frac{R_{HS}}{\theta^n} \qquad \textbf{(3.7.3.3.1)}$$

3.7.5 Application of the Lagrangian-Eulerian Approach for All Interior Nodes and Downstream Boundary Nodes with the Finite Element Method Applied to the Conservative Form of the Reactive Transport Equations for the Upstream Flux Boundaries

3.7.5.1 Fully-Implicit Scheme

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.7.3.1, and the matrix equation for upstream boundary nodes is obtained through the same procedure as that in section 3.7.1.1.

3.7.5.2 Mixed Predictor-Corrector and Operator-Splitting Method

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.7.3.2, and the matrix equation for upstream boundary nodes is obtained through the same procedure as that in section 3.7.1.2.

3.7.5.3 Operator-Splitting Approach

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.7.3.3, and the matrix equation for upstream boundary nodes is obtained through the same procedure as that in section 3.7.1.3.

3.7.6 Application of the Lagrangian-Eulerian Approach for All Interior Nodes and Downstream Boundary Nodes with the Finite Element Method Applied to the Advective Form of the Reactive Transport Equations for the Upstream Flux Boundaries

3.7.6.1 Fully-Implicit Scheme

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.7.3.1, and the matrix equation for upstream boundary nodes is obtained through the same procedure as that in section 3.7.2.1.

3.7.6.2 Mixed Predictor-Corrector and Operator-Splitting Method

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.7.3.2, and the matrix equation for upstream boundary nodes is obtained through the same procedure as that in section 3.7.2.2.

3.7.6.3 Operator-Splitting Approach

For this option, the matrix equation for interior and downstream boundary nodes is obtained through the same procedure as that in section 3.7.3.3, and the matrix equation for upstream boundary nodes is obtained through the same procedure as that in section 3.7.2.3.

3.8 Numerical Implementation of Reactive Transport Coupling among Various Media

This section addresses numerical implement of coupling reactive chemical transport simulations among various media including (1) between 1D river and 2D surface runoff, (2) between 2D surface runoff and 3D subsurface media, (3) between 3D subsurface media and 1D river networks, and (4) among 1D river networks, 2D surface runoff, and 3D subsurface media. For sediment transport simulations, only the coupling between 1D river network and 2D surface runoff is needed, which is similar to the coupling of reactive chemical transport between the two media. Without loss of generality, numerical implementations of coupling for scalar transport (including sediment and kinetic-variable transport) are heuristically given for finite element approximations of the conservative form of transport equations. For Largrangian-Eulerian approximations or finite element approximation of the advective form of transport equations, the implementations of numerical coupling among various media remain valid except care must be taken that the fluxes denote the total fluxes of advective and dispersive/diffusive fluxes.

3.8.1 Coupling between 1D-River and 2D-Overland Water Quality Transport

The interaction between one-dimensional river and two-dimensional surface runoff involves two cases: one is between surface runoff and river nodes (left frame in Fig. 3.8-1) and the other is between surface runoff and junction nodes (right frame in Fig. 3.8-1). For every river node (Node *I* in the left frame of Fig. 3.8-1), there will be associated with two overland nodes (Nodes *J* and *K* in the left frame of Fig. 3.8-1). For every junction node (Node *L* in the right frame of Fig. 3.8-1), there will be associated with a number of overland nodes such as Nodes *J*, *K*, *O*, etc (right frame of Fig. 3.8-1). It should be noted that nodes, such as Nodes *J* and *K* in the right frame of Figure 3.8-1, contribute fluxes to both the river as source/sink of Node *I* and the Junction as source/sink of Node *L*.

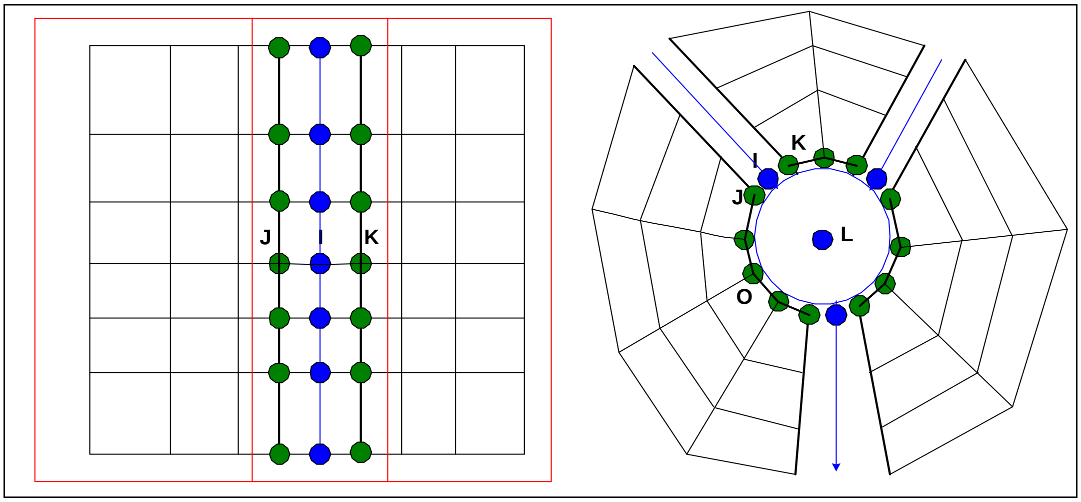

Fig. 3.8-1. Depiction of Interacting River Nodes and Overland Nodes (left) and Junction Node and Overland Nodes (Right)

Numerical approximations of suspended-sediment or kinetic-variable transport equations for one-dimensional river with finite element methods yield the following matrix

$$\begin{bmatrix} -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ C_{I1}^{c} & C_{I2}^{c} & -- & C_{I1}^{c} & -- & -- & C_{IN}^{c} \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \end{bmatrix} \begin{Bmatrix} E_1^{c} \\ E_2^{c} \\ -- \\ E_I^{c} \\ -- \\ -- \\ E_N^{c} \end{Bmatrix} = \begin{Bmatrix} R_1^{c} \\ R_2^{c} \\ -- \\ R_I^{c} \\ -- \\ -- \\ R_N^{c} \end{Bmatrix} + \begin{Bmatrix} M_1^{o1} \\ M_2^{o1} \\ -- \\ M_I^{o1} \\ -- \\ -- \\ M_N^{01} \end{Bmatrix} + \begin{Bmatrix} M_1^{o2} \\ M_2^{o2} \\ -- \\ M_I^{o2} \\ -- \\ -- \\ M_N^{o2} \end{Bmatrix} \qquad \textbf{(3.8.1)}$$

where the superscript c denotes the canal (channel, river, or stream); C_{IJ} is the I-th row, J-th column of the coefficient matrix [C]; E_I denotes the concentration of a suspended sediment or a kinetic variable at Node I; R_I is I-th entry of the load vector {R}; N is the number of nodes in the canal; M_I is the rate of suspended-sediment or kinetic-variable source/sink from (to) the overland flow to (from)canal node I; and the superscripts, $o1$ and o2, respectively, denote canal bank 1 and 2, respectively. Every canal node I involves 3 unknowns, E_I^c, M_I^{o1}, and M_I^{o2}. However, Eq. (3.8.1) gives just one algebraic equation for every canal node I. Clearly, two additional algebraic equations are need for every canal node I. It should be noted that M_I^{o1} and M_I^{o2}denote the following integrations in transforming Eq. (2.5.10) and its initial and boundary conditions or Eq. (2.5.44) and its initial and boundary conditions to Eq. (3.8.1)

$$M_I^{o1} = \int_{X_1}^{X_N} N_I M_{S_n}^{os1} dx \ \ and \ \ M_I^{o2} = \int_{X_1}^{X_N} N_I M_{S_n}^{os2} dx \qquad \textbf{(3.8.2)}$$

for the transport of the n-th suspended-sediment fraction

$$M_I^{o1} = \int_{X_1}^{X_N} N_I M_{E_i}{}^{os1} dx \;\; and \;\; M_I^{o2} = \int_{X_1}^{X_N} N_I M_{E_i}{}^{os2} dx \tag{3.8.3}$$

for the transport of the i-th kinetic variable.

Applications of finite element methods to two-dimensional suspended-sediment or kinetic-variable transport equation yield the following matrix

$$\begin{bmatrix} C_{11}^o & C_{12}^o & -- & -- & -- & -- & C_{1M}^o \\ C_{21}^o & -- & -- & -- & -- & -- & C_{2M}^o \\ -- & -- & -- & -- & -- & -- & -- \\ C_{J1}^o & C_{J2}^o & -- & C_{jj}^o & -- & -- & C_{IM}^o \\ -- & -- & -- & -- & -- & -- & -- \\ C_{K1}^o & C_{K2}^o & -- & -- & C_{KK}^o & -- & C_{KM}^o \\ -- & -- & -- & -- & -- & -- & -- \\ C_{M1}^o & C_{M2}^o & -- & -- & -- & -- & C_{MM}^o \end{bmatrix} \begin{Bmatrix} E_1^o \\ E_2^o \\ -- \\ E_J^o \\ -- \\ E_K^o \\ -- \\ E_M^o \end{Bmatrix} = \begin{Bmatrix} R_1^o \\ R_2^o \\ -- \\ R_J^o \\ -- \\ R_K^o \\ -- \\ R_M^o \end{Bmatrix} - \begin{Bmatrix} -- \\ -- \\ -- \\ M_J^o \\ -- \\ M_K^o \\ -- \\ -- \end{Bmatrix} \tag{3.8.4}$$

where the superscript o denotes the overland; C_{IJ} is the I-th row, J-th column of the coefficient matrix $[C]$; E_I denotes the concentration of suspended sediment or kinetic variable at Node I; R_I is I-th entry of the load vector $\{R\}$; M is the number of nodes in the overland ; and M_J and M_K are the fluxes [M/t] of suspended sediment or kinetic variable from (to) the overland to (from) the canal via nodes J and K, respectively. Equation (3.8.4) indicates that there is one unknown corresponding to one algebraic equation for every interior node. However, for every algebraic equation corresponding to an overland-canal interface node, there are two unknowns, the concentration of suspended sediment or kinetic variable and the sediment or chemical fluxes. Therefore, for every overland-river interface node, one additional equation is needed. Since for every canal node, there are associated two overland-interface nodes, four additional equations are needed for every canal node I for the four additional unknowns M_J^o, M_K^o, M_I^{o1}, and M_I^{o2}.

Before we proceed further, let us refresh ourselves that M_J^o and M_K^o denote the following integration in transforming Eq. (2.6.10) and its initial and boundary conditions or Eq. (2.6.46) and its initial and boundary conditions to Eq. (3.8.4)

$$M_J^o = \int_B \mathbf{n} \bullet \left(W_J \mathbf{q} S_n - N_J h \mathbf{K} \bullet \nabla S_n \right) dB \;\; and \;\; M_K^o = \int_B \mathbf{n} \bullet \left(W_K \mathbf{q} S_n - N_K h \mathbf{K} \bullet \nabla S_n \right) dB \tag{3.8.5}$$

for the transport of the n-th suspended-sediment fraction

$$M_J^o = \int_B \mathbf{n} \bullet \left(W_J \mathbf{q} E_i^m - N_J h \mathbf{K} \bullet \nabla E_i^m \right) dB \;\; and \;\; M_K^o = \int_B \mathbf{n} \bullet \left(W_K \mathbf{q} E_i^m - N_K h \mathbf{K} \bullet \nabla E_i^m \right) dB \tag{3.8.6}$$

for the transport of the i-th kinetic variable.

The additional equations are obtained from two interface boundary conditions: one is the continuity of flux and the other is the assumption that the flux of suspended sediments or kinetic variables through the interface node is due mainly to water flow (i.e., advection). Two of the four additional equations are obtained from the interface condition between the canal node *I* and the overland node *J* as

$$M_J^o = M_I^{o1} \quad and \quad M_J^o = Q_J^o \frac{1}{2}\left(\left(1 + sign\left(Q_J^o\right)\right)E_J^o + \left(1 - sign\left(Q_J^o\right)\right)E_I^c\right) \qquad \textbf{(3.8.7)}$$

For suspended sediment transport, E_J^o and E_I^c denote

$$E_J^o = S_{nJ}{}^o \quad and \quad E_I^c = S_{nI}{}^c \qquad \textbf{(3.8.8)}$$

where $S_{nJ}{}^o$ is the concentration of the suspended sediment of the *n*-th fraction at Node *J* in the overland domain and $S_{nI}{}^c$ is the concentration of the suspended sediment of the *n*-th fraction at Node *I* in the canal domain. For the transport of kinetic variables, E_J^o and E_I^c denote

$$E_J^o = E_i^m{}_J^o \quad and \quad E_I^c = E_i^m{}_I^c \qquad \textbf{(3.8.9)}$$

where $E_i^m{}_J^o$ is the concentration of the mobile portion of the *i*-th kinetic variable at Node *J* in the overland domain and $E_i^m{}_I^c$ is the concentration of the mobile portion of the *i*-th kinetic variable at Node *I* in the canal domain.

The other two additional equations are obtained from the interface condition between the canal Node *I* and the overland Node *K* as follows

$$M_K^o = M_I^{o1} \quad and \quad M_K^o = Q_K^o \frac{1}{2}\left(\left(1 + sign\left(Q_K^o\right)\right)E_K^o + \left(1 - sign\left(Q_K^o\right)\right)E_I^c\right) \qquad \textbf{(3.8.10)}$$

The definition of E_K^o is similar to that of E_J^o.

When the direct contribution of suspended sediment or chemicals from the overland regime to a junction node *L* (Fig. 3.8-1) is significant, the mass balance equation can be written as

$$\frac{d\cancel{V}_L E_L}{dt} = \sum_i \Psi_{iL}^i + \sum_{O \in N_O} M_O^o \quad or \quad \sum_i \Psi_{iL}^i + \sum_{O \in N_O} M_O^o = 0 \qquad \textbf{(3.8.11)}$$

where $\cancel{V}_L$ is the volume of the *L*-th junction, Ψ_{iL}^i is the mass flux from the *iL*-th node of *i*-th reach to the *L*-th junction, and M_O^o is the mass flux from the *O*-th node of the overland regime (superscript *o* t represent overland regime). Additional N_O unknowns have been introduced in Equation (3.8.11). For each overland-junction interface node, say *O* (the right frame in Fig. 3.8.1), the finite element equation written out of Eq. (3.8.4) is

$$C_{O1}^o E_1^o + C_{O2}^o E_2^o + .. + C_{OO}^o E_O^o + .. + C_{OM}^o E_M^o = R_O^o - M_O^o \qquad \textbf{(3.8.12)}$$

It is seen that Equation (3.4.17) involves two unknowns, E_O^o and M_O^o. One equation must be supplemented to the finite element equation to close the system. This equation is obtained by formulating fluxes as

$$M_O^o = Q_O^o \frac{1}{2}\left(\left(1 + sign\left(Q_O^o\right)\right)E_O^o + \left(1 - sign\left(Q_O^o\right)\right)E_L\right) \tag{3.8.13}$$

Equations (3.8.11), (3.8.12), and (3.8.13) for a system of equations for the set of unknowns E_L, E_O^o and M_O^o.

3.8.2 Coupling between 2D-Overalnd and 3D-Subsurface Water Quality Transport

The interaction between two-dimensional overland and three-dimensional subsurface water quality transport is not as straightforward as that between 1D-river and 2D-overland regime because the *i*-th kinetic variable in the 2D-voerland is not necessary to have the same set of species as the *i*-th kinetic variable in the 3D-subsurface media. We will assume that there is no exchange of suspended sediment between 2D-overland and 3D-subsurface media. Only exchanges of aqueous-phase species take place. For every subsurface node (Node *J* in Fig. 3.8-2), there will be associated an overland nodes (Node *I* in Fig. 3.8-2).

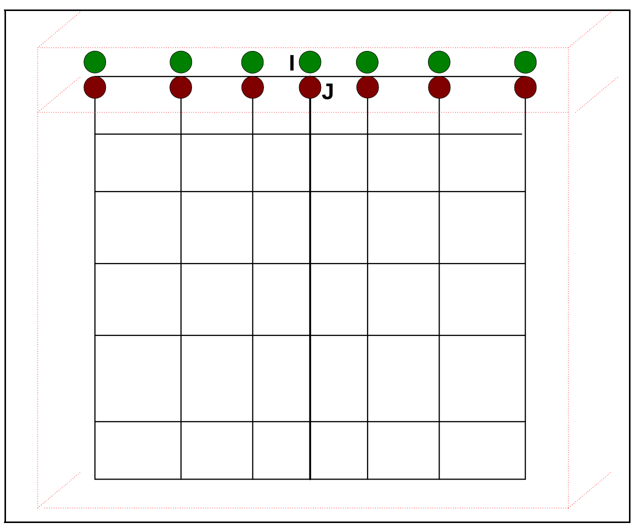

Fig. 3.8-2. Depiction of Interacting Subsurface Nodes and Overland Nodes

Numerical approximations of kinetic-variable transport equation for two-dimensional overland regime with finite element methods yield the following matrix

$$\begin{bmatrix} -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ C_{I1}^{o} & C_{I2}^{o} & -- & C_{II}^{o} & -- & -- & C_{IN}^{o} \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \end{bmatrix} \begin{Bmatrix} E_1^{o} \\ E_2^{o} \\ -- \\ E_I^{o} \\ -- \\ -- \\ E_N^{o} \end{Bmatrix} = \begin{Bmatrix} R_1^{o} \\ R_2^{o} \\ -- \\ R_I^{o} \\ -- \\ -- \\ R_N^{o} \end{Bmatrix} + \begin{Bmatrix} M_1^{io} \\ M_2^{io} \\ -- \\ M_I^{io} \\ -- \\ -- \\ M_N^{io} \end{Bmatrix} \quad \textbf{(3.8.14)}$$

where the superscript o denotes the overland; C_{IJ} is the I-th row, J-th column of the coefficient matrix $[C]$; E_I denotes the concentration of a kinetic variable at Node I; R_I is I-th entry of the load vector $\{R\}$; N is the number of nodes in the overland; and M_I is the rate of the kinetic-variable source/sink from (to) the subsurface to (from) the overland node I (the superscript, io, denotes the exfiltration from subsurface media to overland). Every overland node I involves two unknowns, E_I^o, and M_I^{io}. However, Eq. (3.8.14) gives just one algebraic equation for every canal node I. Clearly, one additional algebraic equation is need for every overland node I. To formulate this equation, it is noted that, for the i-th overland kinetic variable, M_I^{io} is the source/sink rate of the i-th kinetic variable at the I-th node due to infiltration (negative value) or exfitration (positive value). This equation is obtained as follows

$$M_I^{io} = (Q_I^{io})\frac{1}{2}\left(\left(1+sign\left(Q_I^{io}\right)\right)\sum_{j\in M_a} a_{ij}^o C_{jJ}^s + \left(1-sign\left(Q_I^{io}\right)\right)\sum_{j\in M_a} a_{ij}^o C_{jI}^o\right) \quad \textbf{(3.8.15)}$$

where M_a is the set of aqueous species, a_{ij}^o is the ij-th entry of the decomposed unit matrix via diagonalization of the reaction network in the overland domain, C_{jJ}^s is the concentration of the j-th subsurface species at the J-th node of the subsurface domain, and C_{jI}^o is the concentration of the j-th overland species at the I-th node of the overland domain.

Applications of finite element methods to three-dimensional kinetic-variable transport equations for subsurface media yield the following matrix

$$\begin{bmatrix} C_{11}^{s} & C_{12}^{s} & -- & -- & -- & -- & C_{1M}^{s} \\ C_{21}^{s} & -- & -- & -- & -- & -- & C_{2M}^{s} \\ -- & -- & -- & -- & -- & -- & -- \\ C_{J1}^{s} & C_{J2}^{s} & -- & C_{JJ}^{s} & -- & -- & C_{IM}^{s} \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ C_{M1}^{s} & C_{M2}^{s} & -- & -- & -- & -- & C_{MM}^{s} \end{bmatrix} \begin{Bmatrix} E_1^{s} \\ E_2^{s} \\ -- \\ E_J^{s} \\ -- \\ -- \\ E_M^{s} \end{Bmatrix} = \begin{Bmatrix} R_1^{s} \\ R_2^{s} \\ -- \\ R_J^{s} \\ -- \\ -- \\ R_M^{s} \end{Bmatrix} - \begin{Bmatrix} -- \\ -- \\ -- \\ M_J^{s} \\ -- \\ -- \\ -- \end{Bmatrix} \quad \textbf{(3.8.16)}$$

where the superscript s denotes the subsurface media; C_{IJ} is the I-th row, J-th column of the coefficient matrix [C]; E_J denotes the temperature or salinity at Node J; R_J is J-th entry of the load vector {R}; M is the number of nodes in the subsurface; and M_J is the rate of thermal or salt sink/source from/to the subsurface node J to/from the corresponding overland node I. Equation (3.8.15) indicates that there is one unknown corresponding to one algebraic equation for every interior node. However, for every algebraic equation corresponding to a subsurface-overland interface node, there are two unknowns, the concentration of the i-th subsurface kinetic variable at node J, E_J^s, and its flux, M_J^s. Therefore, one additional equation is needed. This equation is obtained from

$$M_J^s = (Q_J^s)\frac{1}{2}\left(\left(1+sign\left(Q_J^s\right)\right)\sum_{j\in M_a} a_{ij}^s C_{jJ}^s + \left(1-sign\left(Q_J^s\right)\right)\sum_{j\in M_a} a_{ij}^s C_{jI}^o\right) \quad \textbf{(3.8.17)}$$

where a_{ij}^s is the ij-th entry of the decomposed unit matrix via diagonalization of the reaction network in the subsurface media.

3.8.3 Coupling between 3-D Subsurface and 1-D Surface Flows

The interaction between three-dimensional subsurface and one-dimensional river water quality transport involves three options: (1) river is discretized as finite-width and finite-depth on the three-dimensional subsurface media (Fig. 3.8-3), (2) river is discretized as finite-width and zero-depth on the three-dimensional subsurface media (Fig. 3.4-4), and (3) river is discretized as zero-width and zero-depth on the three-dimensional subsurface media (Fig. 3.4-5). Option 1 is the most realistic one. However, because of the computational demands, it is normally used in small scale studies involving the investigations of infiltration and discharge between river and subsurface media on a local scale. Option 2 is normally used in medium scale studies while Option 3 is normally employed in large scale investigations. In Option 1, for every river node there are associated with a number of subsurface interfacing nodes such as K, .., J, .., and L(Fig. 3.8-3). In Option 2, for every river node there are associated with three subsurface interfacing nodes K, J, and L (Fig. 3.8-4). In Option 3, for every river node there is associated with one subsurface interfacing node J (Fig. 3.8-5).

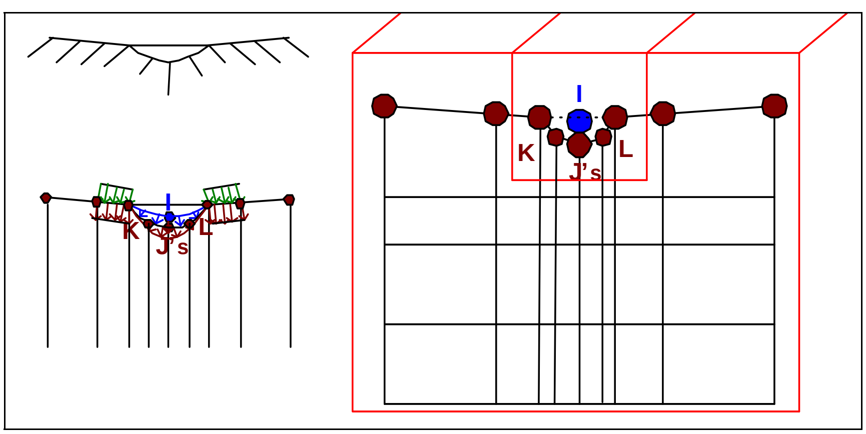

Fig. 3.8-3. Rivers Are Discretized as Finite-Width and Finite-Depth on the Subsurface Media

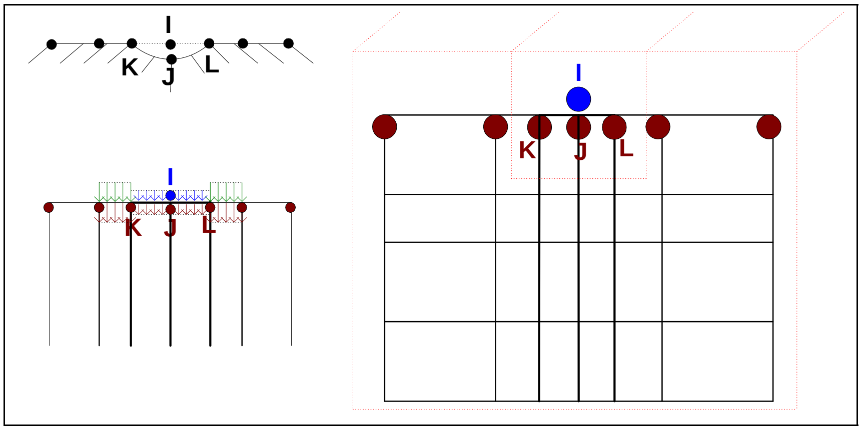

Fig. 3.8-4. Rivers Are Discretized as Finite-Width and Zero-Depth on the Subsurface Media

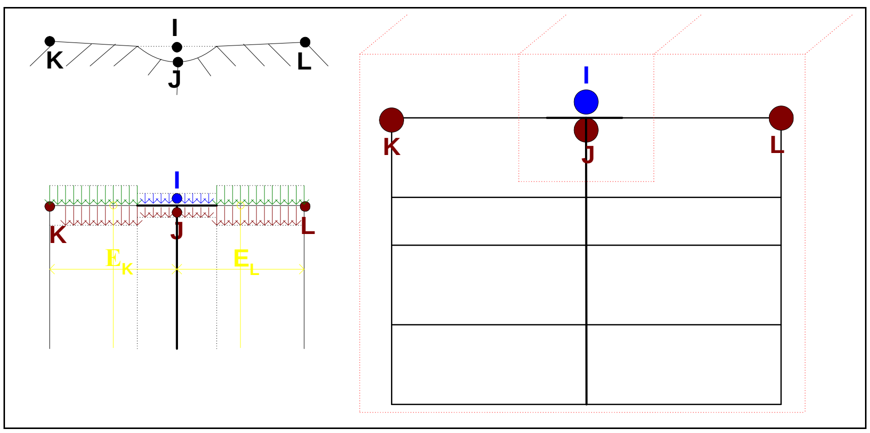

Fig. 3.8-5. Rivers Are Discretized as Zero-Width and Zero-Depth on the Subsurface Media

Numerical approximations of i-th kinetic-variable transport equation for one-dimensional river with finite element methods yield the following matrix

$$\begin{bmatrix} -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ C_{I1}^{c} & C_{I2}^{c} & -- & C_{II}^{c} & -- & -- & C_{IN}^{c} \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \\ -- & -- & -- & -- & -- & -- & -- \end{bmatrix} \begin{Bmatrix} E_1^c \\ E_2^c \\ -- \\ E_I^c \\ -- \\ -- \\ E_N^c \end{Bmatrix} = \begin{Bmatrix} R_1^c \\ R_2^c \\ -- \\ R_I^c \\ -- \\ -- \\ R_N^c \end{Bmatrix} + \begin{Bmatrix} M_1^{ic} \\ M_2^{ic} \\ -- \\ M_I^{ic} \\ -- \\ -- \\ M_N^{ic} \end{Bmatrix} \quad \textbf{(3.8.18)}$$

where the superscript c denotes the canal (channel, river, or stream); C_{IJ} is the I-th row, J-th column of the coefficient matrix [C]; E_I denotes the temperature or salinity at Node I; R_I is I-th entry of the load vector {R}; N is the number of nodes in the canal; and M_I^{ic} is the mass rate of the kinetic-variable source/sink from (to) the subsurface to (from) canal node I due to infiltration/exfiltration. Every canal node I involves two unknowns, E_I^c and M_I^{ic}. However, Eq. (3.8.18) gives just one algebraic equation for every canal node I. Clearly, one additional algebraic equation is need for every canal node I.

For example, taking Option 2 where there are three nodes associated with one canal node, the applications of finite element methods to three-dimensional kinetic-variable transport equation in the subsurface media yields

$$\begin{bmatrix} C_{11}^s & C_{12}^s & -- & -- & -- & -- & C_{1M}^s \\ C_{21}^s & -- & -- & -- & -- & -- & C_{2M}^s \\ -- & -- & -- & -- & -- & -- & -- \\ C_{K1}^s & C_{K2}^s & C_{KK}^s & -- & -- & -- & C_{KM}^s \\ C_{J1}^s & C_{J2}^s & -- & C_{JJ}^s & -- & -- & C_{1M}^s \\ C_{L1}^s & C_{L2}^s & C_{<<}^s & -- & -- & -- & C_{LM}^s \\ -- & -- & -- & -- & -- & -- & -- \\ C_{M1}^s & C_{M2}^s & -- & -- & -- & -- & C_{MM}^s \end{bmatrix} \begin{Bmatrix} E_1^s \\ E_2^s \\ -- \\ E_K^s \\ E_J^s \\ E_L^s \\ -- \\ E_M^s \end{Bmatrix} = \begin{Bmatrix} R_1^s \\ R_2^s \\ -- \\ R_K^s \\ R_J^s \\ R_L^s \\ -- \\ R_M^s \end{Bmatrix} - \begin{Bmatrix} -- \\ -- \\ -- \\ M_K^s \\ M_J^s \\ M_L^s \\ -- \\ -- \end{Bmatrix} \quad \textbf{(3.8.19)}$$

where the superscript s denotes the subsurface media; C_{IJ} is the I-th row, J-th column of the coefficient matrix [C]; E_J denotes the temperature or salinity at Node J; R_J is J-th entry of the load vector {R}; M is the number of nodes in the overland ; and M_K, M_J and M_L are the rates of thermal or salt sink/source from/to the subsurface water to/from the canal via nodes K, J and L, respectively. Equation (3.8.19) indicates that there is one unknown corresponding to one algebraic equation for every interior node. However, for every algebraic equation corresponding a subsurface-canal interface node, there are two unknowns, concentration of the kinetic variable and its flux. Therefore, for every subsurface-river interface node, one additional equation is needed. Since for every canal node, there are associated three subsurface-interface nodes, four additional equations are needed for every canal node I for the four additional unknowns M_I^{ic}, M_K^s, M_J^s, and M_L^s.

These four additional equations are obtained with the assumptions that only aqueous species are involved in the exchange between the canal node I and the subsurface nodes K, J, and L and the exchange is mainly due to advection. These assumptions result in the following four equations:

$$M_I^{ic} = \frac{1}{2}(Q_I^{ic})\left(\left(1 - sign\left(Q_I^{ic}\right)\right)\sum_{j \in M_a} a_{ij}^c C_{jI}^c\right) + \frac{1}{2}\left(1 + sign(Q_I^{ic})\right) \times$$
$$\left(Q_K^s \sum_{j \in M_a} a_{ij}^c C_{jK}^s + Q_J^s \sum_{j \in M_a} a_{ij}^c C_{jJ}^s + Q_L^s \sum_{j \in M_a} a_{ij}^c C_{jL}^s - Q_K^{rains} \sum_{j \in M_a} a_{ij}^c C_{jK}^{rain} - Q_L^{rains} \sum_{j \in M_a} a_{ij}^c C_{jL}^{rain}\right) \quad \textbf{(3.8.20)}$$

$$M_J^s = \frac{1}{2}(Q_J^s)\left(\left(1-sign\left(Q_J^s\right)\right)\sum_{j\in M_a} a_{ij}^s C_{jI}^c\right) + \frac{1}{2}(Q_J^s)\left(\left(1+sign\left(Q_J^s\right)\right)\sum_{j\in M_a} a_{ij}^s C_{jJ}^s\right) \quad \textbf{(3.8.21)}$$

$$M_K^s = \frac{1}{2}(Q_K^s)\left(\left(1-sign\left(Q_K^s\right)\right)\sum_{j\in M_a} a_{ij}^s C_{jI}^c\right) + \frac{1}{2}(Q_K^s)\left(\left(1+sign\left(Q_K^s\right)\right)\sum_{j\in M_a} a_{ij}^s C_{jK}^s\right) \quad \textbf{(3.8.22}$$

$$M_L^s = \frac{1}{2}(Q_L^s)\left(\left(1-sign\left(Q_L^s\right)\right)\sum_{j\in M_a} a_{ij}^s C_{jI}^c\right) + \frac{1}{2}(Q_L^s)\left(\left(1+sign\left(Q_L^s\right)\right)\sum_{j\in M_a} a_{ij}^s C_{jL}^s\right) \quad \textbf{(3.8.23)}$$

where M_a is the set of aqueous species, a_{ij}^c is the *ij*-th entry of the decomposed unit matrix via diagonalization of the reaction network in the canal domain, C_{jI}^c is the concentration of the *j*-th canal species at the I-th node of the canal domain, C_{jJ}^s is the concentration of the *j*-th subsurface species at the *J*-th node of the subsurface domain, C_{jK}^s is the concentration of the *j*-th subsurface species at the *K*-th node of the subsurface domain, C_{jL}^s is the concentration of the *j*-th subsurface species at the *L*-th node of the subsurface domain, C_{jK}^{rain} is the concentration of the *j*-th species of the rainfall that falls on the *K*-th node of the subsurface domain, C_{jL}^{rain} is the concentration of the *j*-th species of the rainfall that falls on the *L*-th node of the subsurface domain, and a_{ij}^s is the *ij*-th entry of the decomposed unit matrix via diagonalization of the reaction network in the subsurface domain.

3.8.4 Coupling Among River, Overland, and Subsurface Flows

The interaction among one-dimensional river, two-dimensional overland, and three-dimensional subsurface flows involves three options: (1) river is discretized as finite-width and finite-depth on the three-dimensional subsurface media (Fig. 3.8-6), (2) river is discretized as finite-width and zero-depth on the three-dimensional subsurface media (Fig. 3.8-7), and (3) river is discretized as zero-width and zero-depth on the three-dimensional subsurface media (Fig. 3.4-8). Option 1 is the most realistic one. However, because of the computational demands, it is normally used in small scale studies involving the investigations of infiltration and discharge between river and subsurface media on a local scale. Option 2 is normally used in medium scale studies while Option 3 is normally employed in large scale investigations. In Option 1, for every river node there are associated with two overland nodes *M* and *N* and a number of subsurface interfacing nodes such as *K*. , *J*, .., and *L* (Fig. 3.8-6). In Option 2, for every river node *I*, there are associated with two overland nodes *M* and *N* and three subsurface interfacing nodes *K*, *J*, and *L* (Fig. 3.4-7). In Option 3, for every river node *I*, there is associated with two overland nodes *M* and *N* one subsurface node *J* (Fig. 3.8-8).

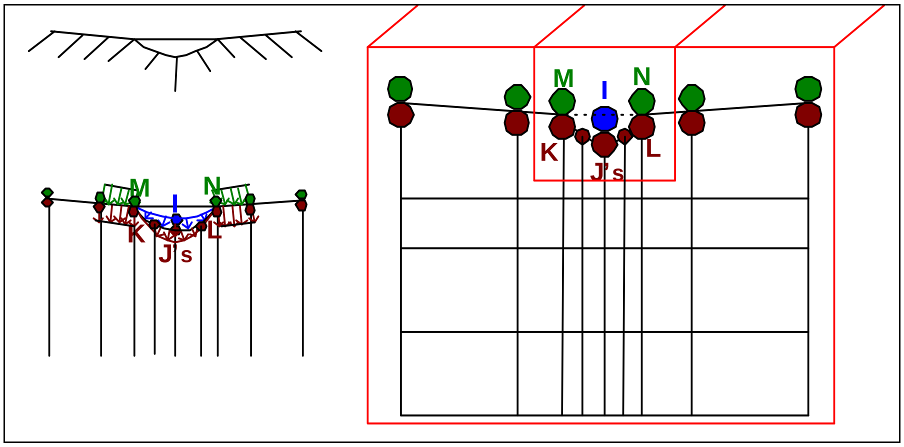

Fig. 3.8-6. Interfacing Nodes for Every River Node when Rivers Are Discretized as Finite-Width and Finite-Depth

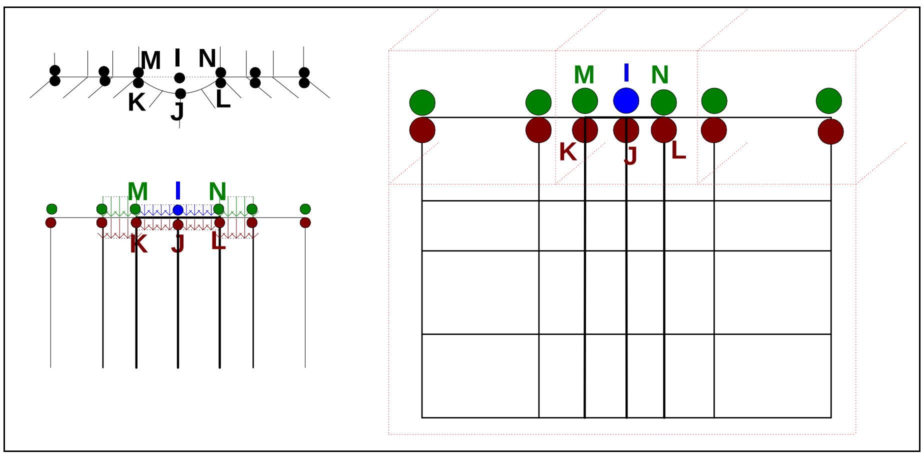

Fig. 3.8-7. Interfacing Nodes for Every River Node when Rivers Are Discretized as Finite-Width and Zero-Depth

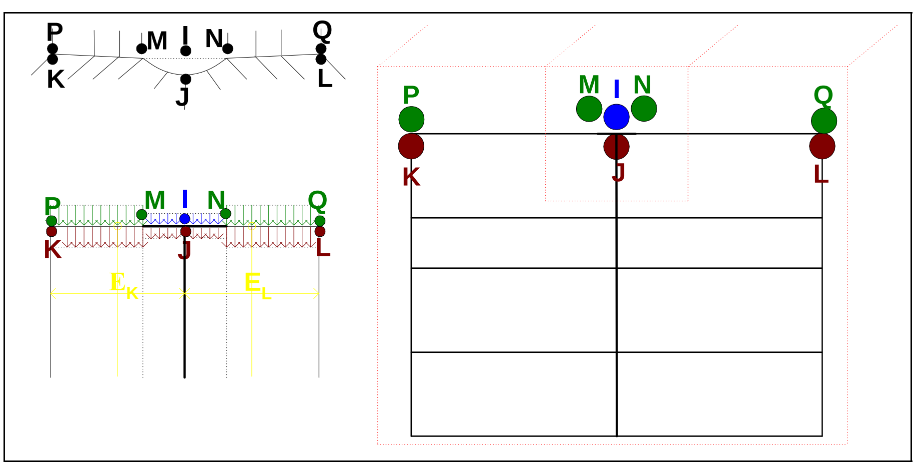

Fig. 3.8-8. Interfacing Nodes for Every River Node when Rivers Are Discretized as Zero-Width and Zero-Depth

Similar to the coupling of salt transport among river, overland, and subsurface media, the coupling of water quality transport is achieved by imposing the continuity of water quality fluxes and formulation of individual node fluxes.

Interaction between Overland Node *M* and Canal Node *I*. Two equations are obtained based on the continuity of fluxes and the formulation of fluxes as

$$M_I^{o1} = Q_I^{o1} \frac{1}{2}\left(\left(1 + sign\left(Q_I^{o1}\right)\right)E_M^o + \left(1 - sign\left(\left(Q_I^{o1}\right)\right)E_I^c\right) \quad and$$
$$M_M^o = Q_M^o \frac{1}{2}\left(\left(1 + sign\left(Q_M^o\right)\right)E_M^o + \left(1 - sign\left(\left(Q_M^o\right)\right)E_I^c\right) \tag{3.8.24}$$

Interaction between Overland Node *N* and Canal Node *I*. Two equations are obtained based on the continuity of fluxes and the formulation of fluxes as

$$M_I^{o2} = Q_I^{o2} \frac{1}{2}\left(\left(1 + sign\left(Q_I^{o2}\right)\right)E_N^o + \left(1 - sign\left(\left(Q_I^{o2}\right)\right)E_I^c\right) \quad and$$
$$M_N^o = Q_N^o \frac{1}{2}\left(\left(1 + sign\left(Q_N^o\right)\right)E_N^o + \left(1 - sign\left(\left(Q_N^o\right)\right)E_I^c\right) \tag{3.8.25}$$

Interaction between Overland Node *M*, Subsurface Node *K*, and Canal Node *I*. Two equations are obtained based on the continuity of fluxes and the formulation of fluxes as

$$M_M^{io} = \left\{\frac{1}{2}\left(1 - sign\left(Q_M^{io}\right)\right)Q_M^{io}\sum_{j\in M_a} a_{ij}^o C_{jM}^o + \frac{1}{2}\left(1 + sign\left(Q_M^{io}\right)\right)\left(Q_K^s \sum_{j\in M_a} a_{ij}^o C_{jK}^s - \frac{1}{4}Q_I^{ic}\sum_{j\in M_a} a_{ij}^o C_{jI}^c E_I^c\right)\right\}$$
$$and \tag{3.8.26}$$
$$M_K^s = \left\{\frac{1}{2}\left(1 + sign\left(Q_K^s\right)\right)Q_K^s \sum_{j\in M_a} a_{ij}^s C_{jK}^s + \frac{1}{2}\left(1 - sign\left(Q_K^s\right)\right)\left(Q_M^{io}\sum_{j\in M_a} a_{ij}^s C_{jM}^o + \frac{1}{4}Q_I^{ic}\sum_{j\in M_a} a_{ij}^s C_{jI}^c\right)\right\}$$

where M_a is the set of aqueous species, a_{ij}^o is the *ij*-th entry of the decomposed unit matrix via diagonalization of the reaction network in the overland domain.

Interaction between River Bank Node *N*, Subsurface Node *L*, and Canal Node *I*. Two equations are obtained based on the continuity of fluxes and the formulation of fluxes

$$M_N^{io} = \left\{\frac{1}{2}\left(1 - sign\left(Q_N^{io}\right)\right)Q_N^{io}\sum_{j\in M_a} a_{ij}^o C_{jN}^o + \frac{1}{2}\left(1 + sign\left(Q_N^{io}\right)\right)\left(Q_L^s \sum_{j\in M_a} a_{ij}^o C_{jL}^s - \frac{1}{4}Q_I^{ic}\sum_{j\in M_a} a_{ij}^o C_{jI}^c\right)\right\}$$
$$and \tag{3.8.27}$$
$$M_L^s = \left\{\frac{1}{2}\left(1 + sign\left(Q_L^s\right)\right)Q_L^s \sum_{j\in M_a} a_{ij}^s C_{jL}^s + \frac{1}{2}\left(1 - sign\left(Q_L^s\right)\right)\left(Q_N^{io}\sum_{j\in M_a} a_{ij}^s C_{jN}^o + \frac{1}{4}Q_I^{ic}\sum_{j\in M_a} a_{ij}^s C_{jI}^c\right)\right\}$$

Interaction between Subsurface Node *J* and Canal Node *I*. Two equations are obtained based on the continuity of fluxes and the formulation of fluxes as

$$
\begin{aligned}
M_I^{ic} &= \left(\frac{1}{2}\left(1+sign\left(Q_I^{ic}\right)\right) 2Q_J^s \sum_{j\in M_a} a_{ij}^c C_{jJ}^s + \frac{1}{2}\left(1-sign\left(Q_I^{ic}\right)\right) Q_I^{ic} \sum_{j\in M_a} a_{ij}^c C_{jI}^c \right) \quad and \\
M_J^s &= \left(\frac{1}{2}\left(1+sign\left(Q_J^s\right)\right) Q_J^s \sum_{j\in M_a} a_{ij}^s C_{jJ}^s + \frac{1}{2}\left(1-sign\left(Q_J^s\right)\right) \frac{1}{2} Q_I^{ic} \sum_{j\in M_a} a_{ij}^s C_{jI}^c \right)
\end{aligned}
\qquad \textbf{(3.8.28)}
$$

3.9 Vastly Different Time Scales among Various Media

The time scales for hydrology and hydraulics and water quality transport in river/stream/canal networks, overland regime and subsurface media are vastly different. The time scale for flow and transport may be in the order of seconds and minutes in 1D-river/stream/canal networks, minutes in 2D-overland regime, and hours, days or even weeks in 3D-subsurface media. To handle this kind of very different time-scale problems, the approach of using variable time-step sizes among different domains is taken. Figure 3.9-1 shows the model structure of over-all coupling between various interfacial media. In Figure 3.9-1, Δt = GT is the global time-step size (it is noted that total simulation time may consist of several Δt's); GTS is the number of time steps in each GT and Δt_{GT} is the time-step size; 3DF is the number of time steps for 3D flow simulations in each GT and Δt_{3DF} is time step size; 2DF is the number of time steps for 2D flow simulations and Δt_{2DF} is the time step size; 1DF is the number of time steps for 1D flow simulations and Δt_{1DF} is the time step size.

Figures 3.9-2 shows the detail structure on 1D only river/stream/canal networks simulations. For flow computation in one time step, we first linearize all coefficients in and boundary conditions (by linearize boundary conditions, we mean, for example, to fix variable-type boundary conditions if they are prescribed) for the governing equations using previous iterates and solve the linearized equations within the nonlinear loop. Within the nonlinear loop, first solve flow equations to obtain HQW1, where HQW1 is the water depth and discharge for the 1D case; then for every several flow time steps, solve salinity and thermal transport equation to yield C1 and T1, where C1 and T1 are the salt concentration and temperature, respectively. When fluid flow and salt and thermal transport are solved to convergences, repeat one more nonlinear loop to provide flow fields (i.e., HQW1) for the simulation of reactive chemical transport. The solution of reactive chemical transport would render CR1, where CR1 is the concentration of reactive biogeochemical species for 1D. After density-dependent flow fields, salinity, temperature, and reactive chemical transport are solved, proceed to the next time step. Figures 3.9-3 and 3.9-4 show detail computational structures for simulations in 2D overland and 3D subsurface media, respectively.

Figures 3.9-5, 3.9-6, and 3.9-7 show detail structures for simulating in coupled 1D and 2D, coupled 2D and 3D, and coupled 3D and 1D flow and transport, respectively. In all eight figures, the naming convention of the state-variables is systematic combination of H, Q, C, T, CR, R, W, P, 0, 1, 2, and 3. H denotes water depth or head, Q denotes discharge, C denote salt concentration, T denote temperature, CR denote concentration of reactive entities, R denotes source/sinks, W denotes working iterative values, P denotes previous time, 0 denote initial values, 1 denote 1D, 2 denotes 2D, and 3 denotes 3D. For example, HQW1 (at convergence, HQW1 would be HQ1) is the water depth and discharge of the iterative working values for 1D case; CR2 is the concentrations of reactive entities for 2D cases; TP1 is the temperature at the previous time step for 1D cases. DIV denotes the divergence of the velocity, i.e. DIV = $\nabla \cdot \mathbf{V}$.

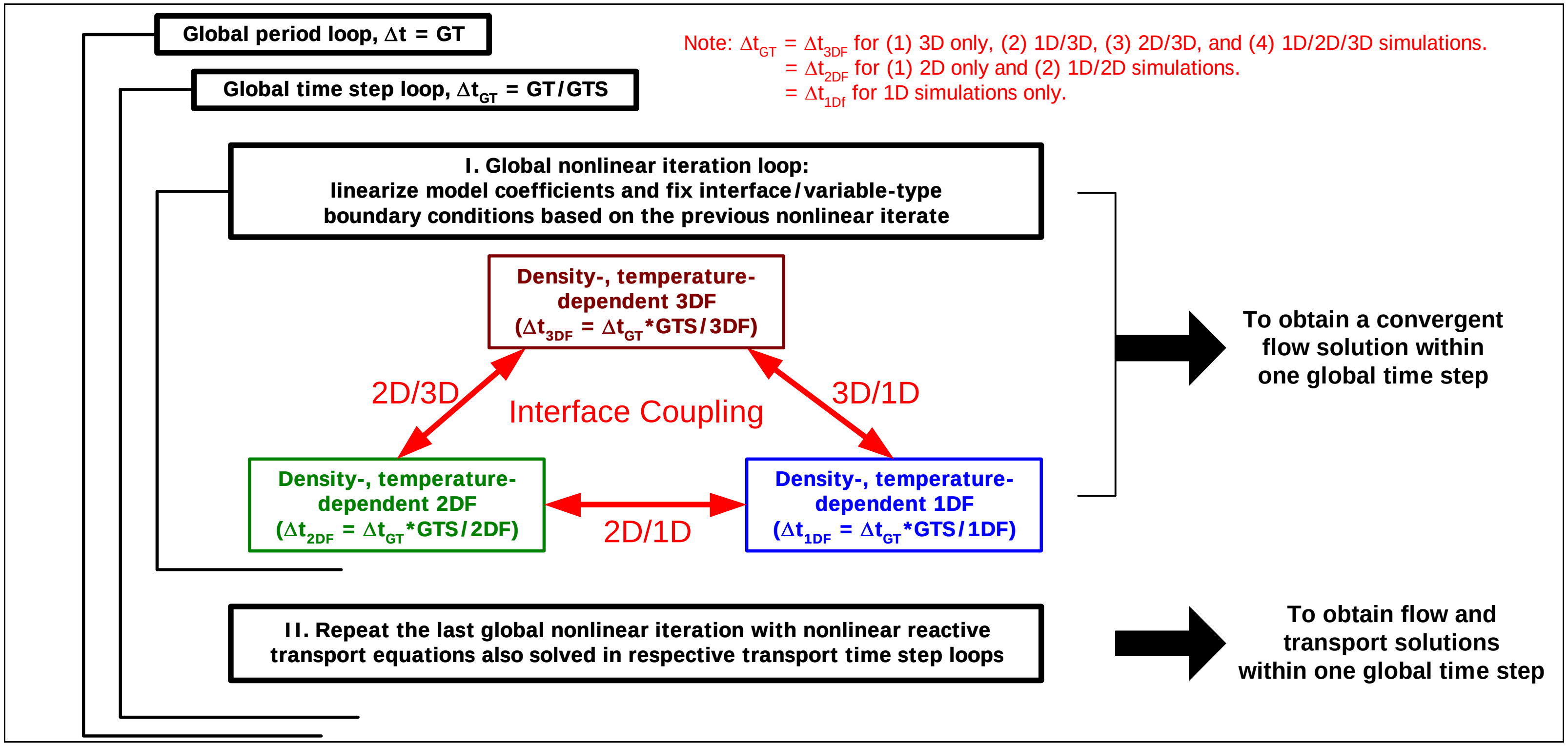

Fig. 3.9-1. Overall Coupled Structure of WASH123D

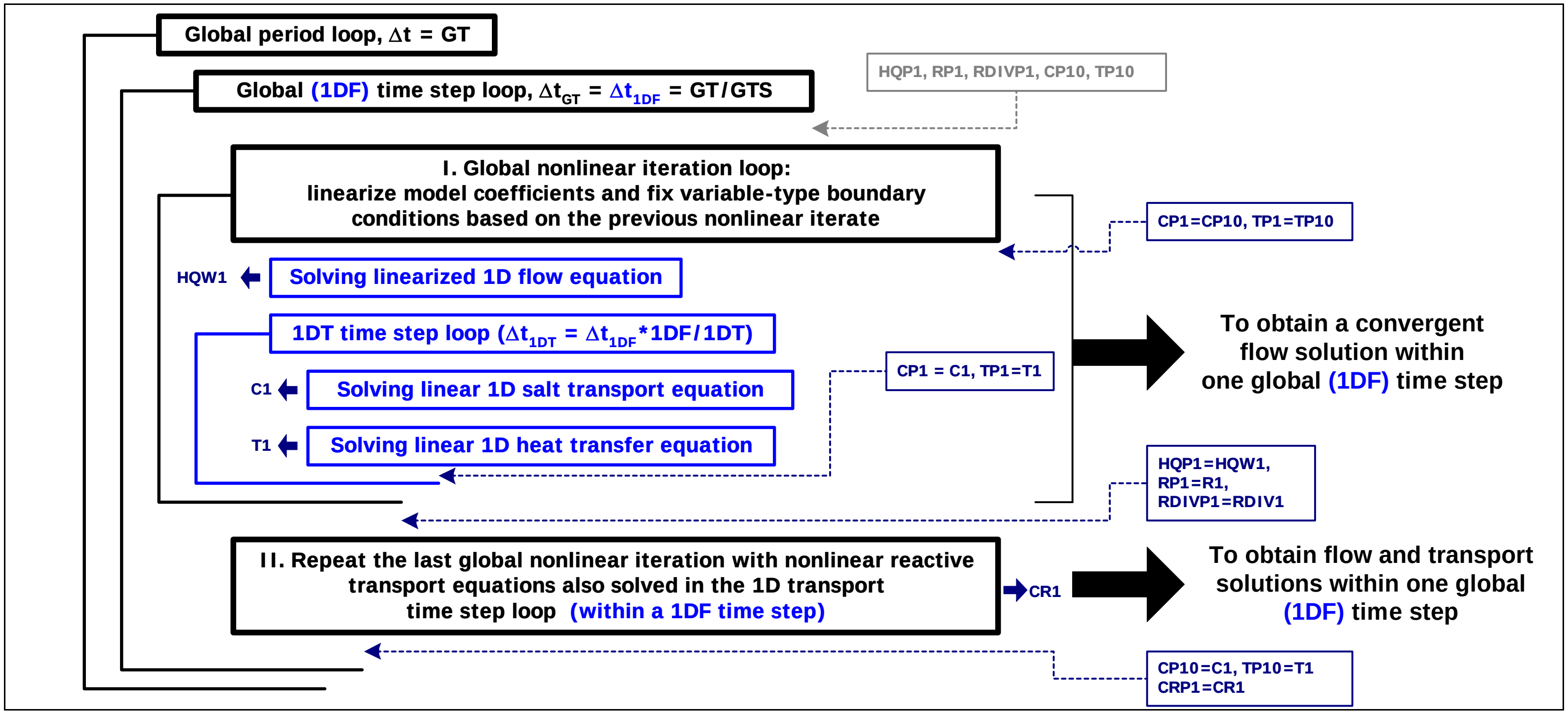

Fig. 3.9-2. Computation Structure of WASH123D for 1D only Simulations

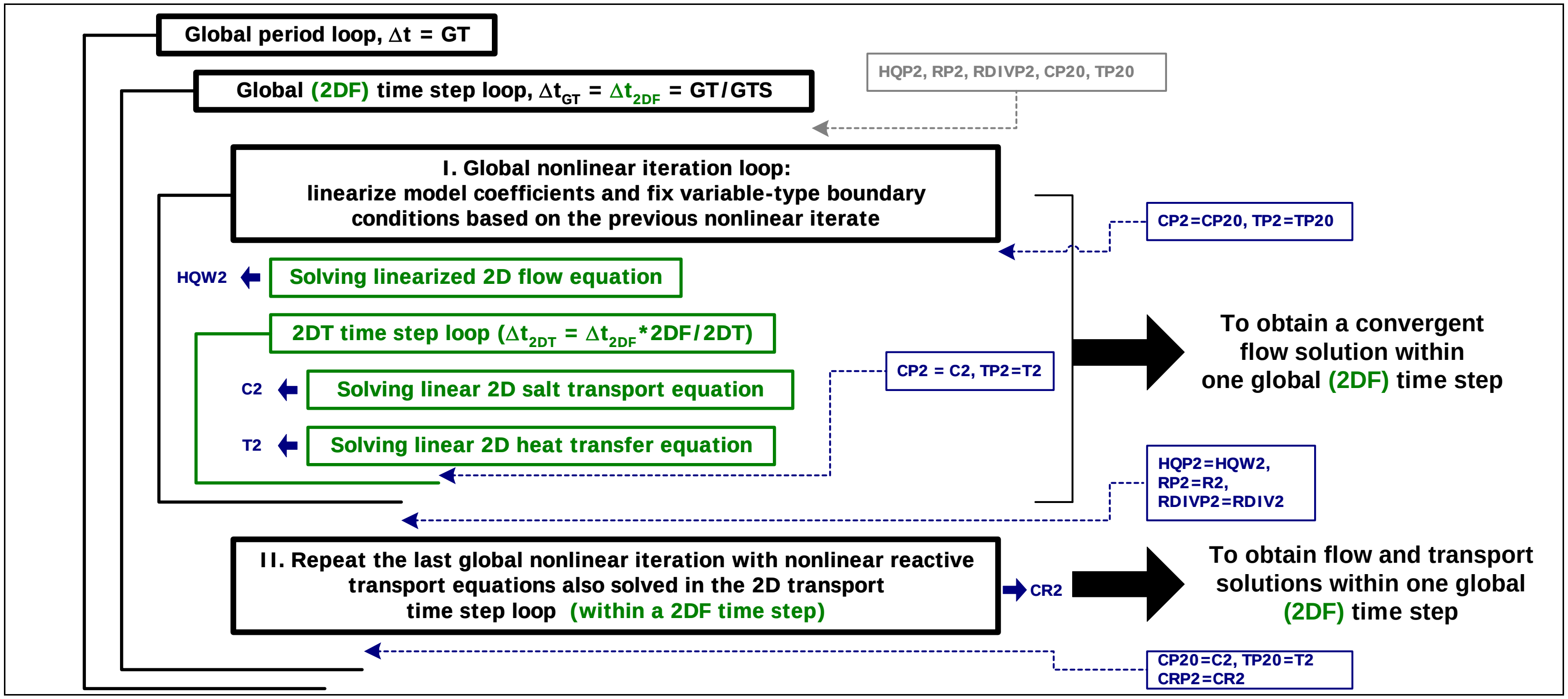

Fig. 3.9-3. Computation Structure of WASH123D for 2D only Simulations

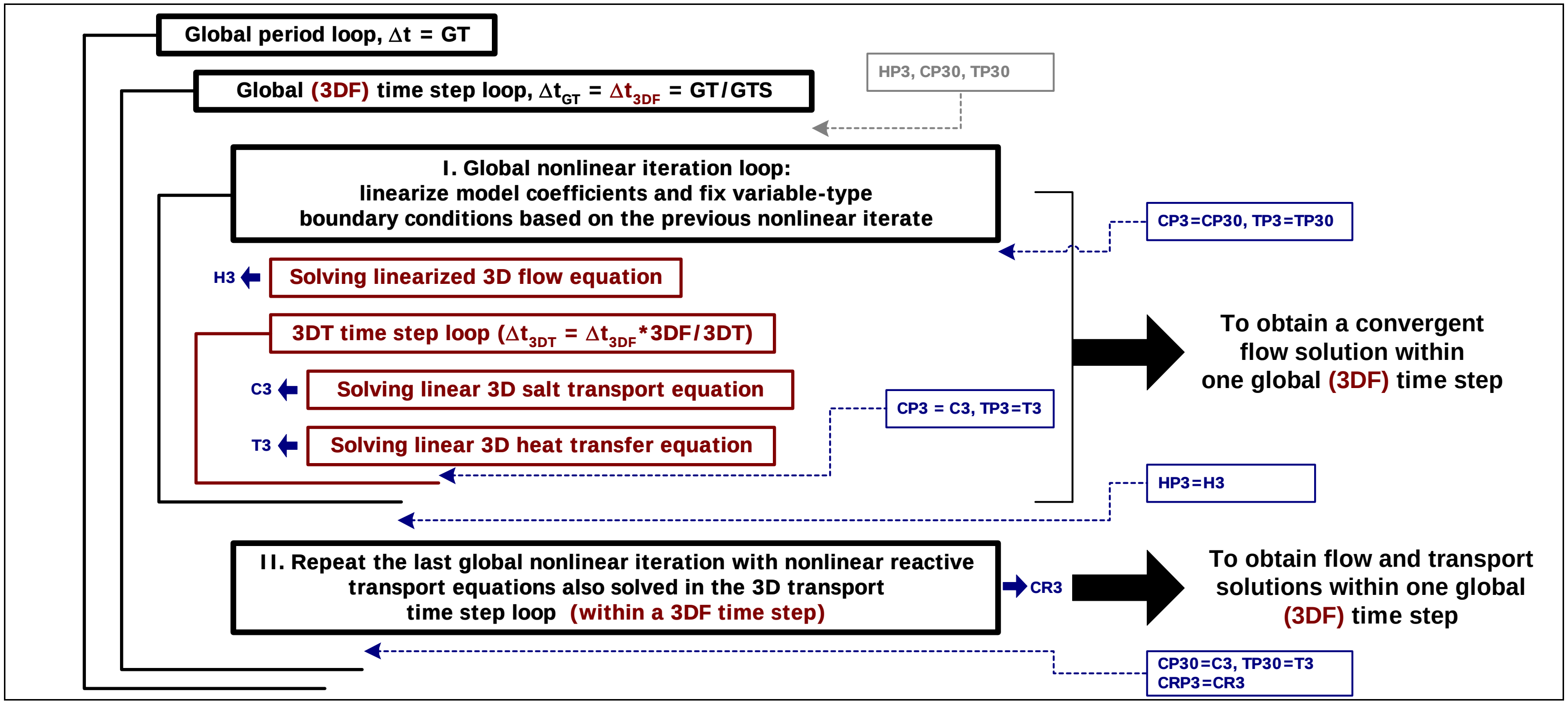

Fig. 3.9-4. Computation Structure of WASH123D for 3D only Simulations

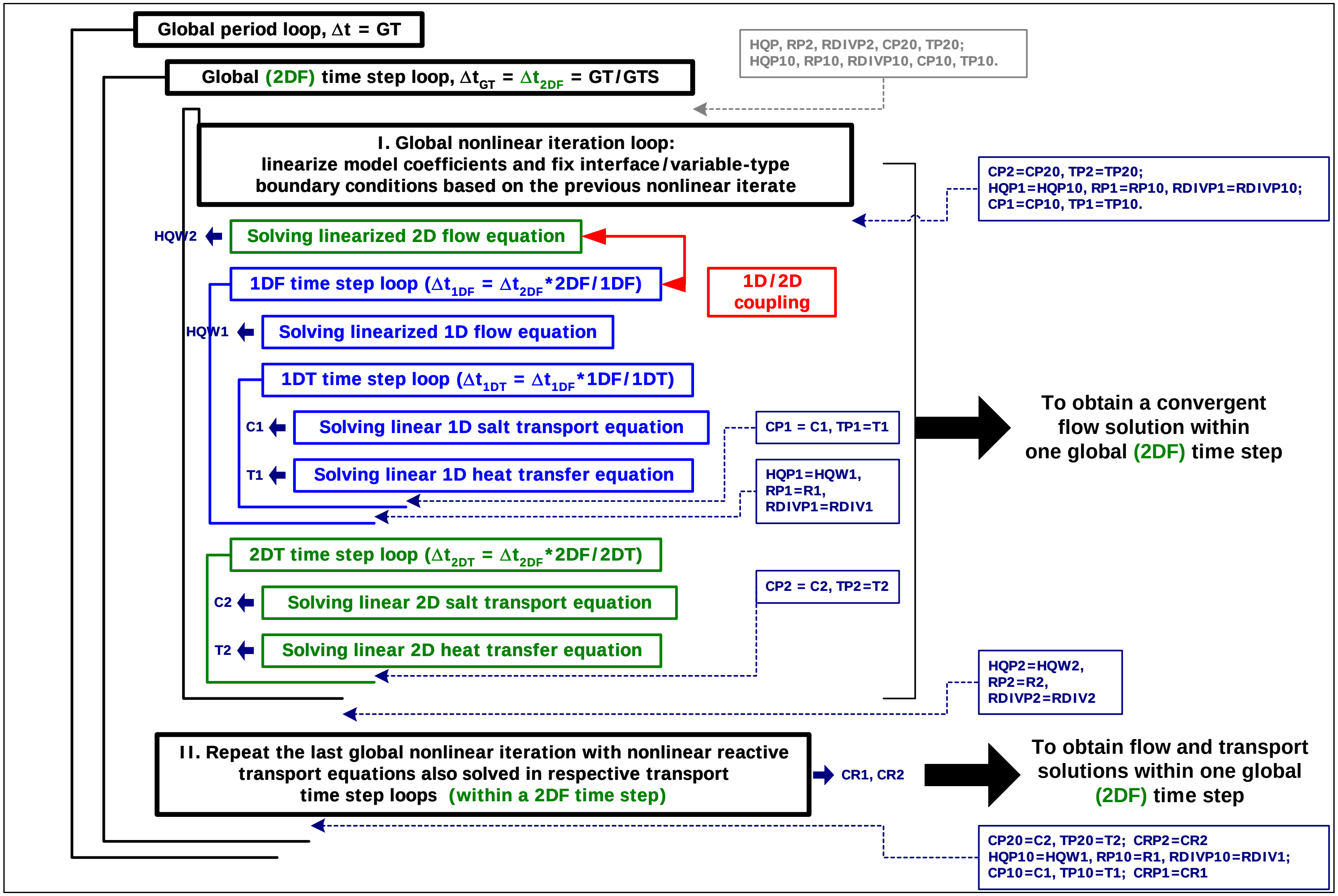

Fig. 3.9-5. Computation Structure of WASH123D for Coupled 1D/2D Simulations

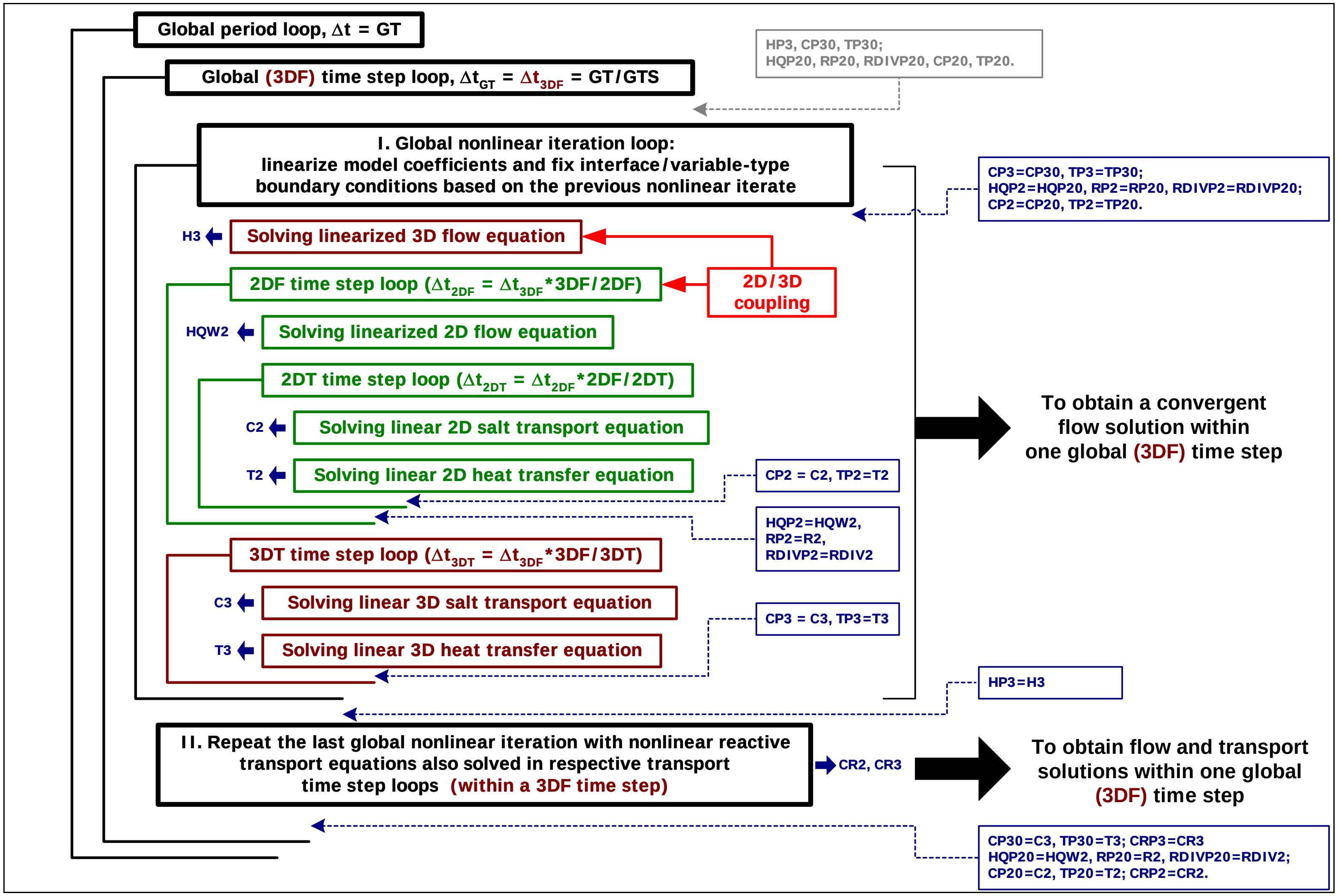

Fig. 3.9-6. Computation Structure of WASH123D for Coupled 2D/3D Simulations

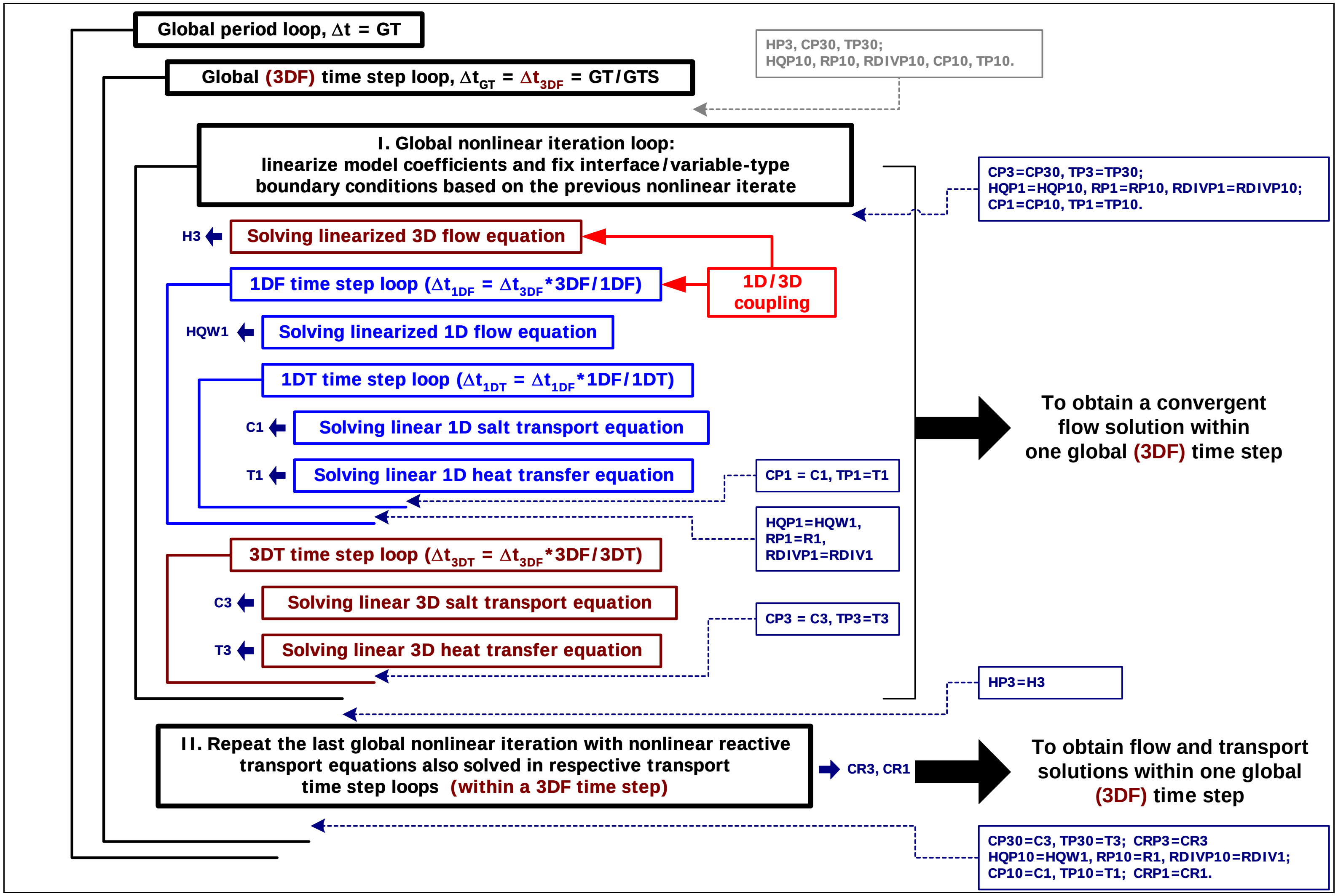

Fig. 3.9-7. Computation Structure of WASH123D for Coupled 3D/1D Simulations

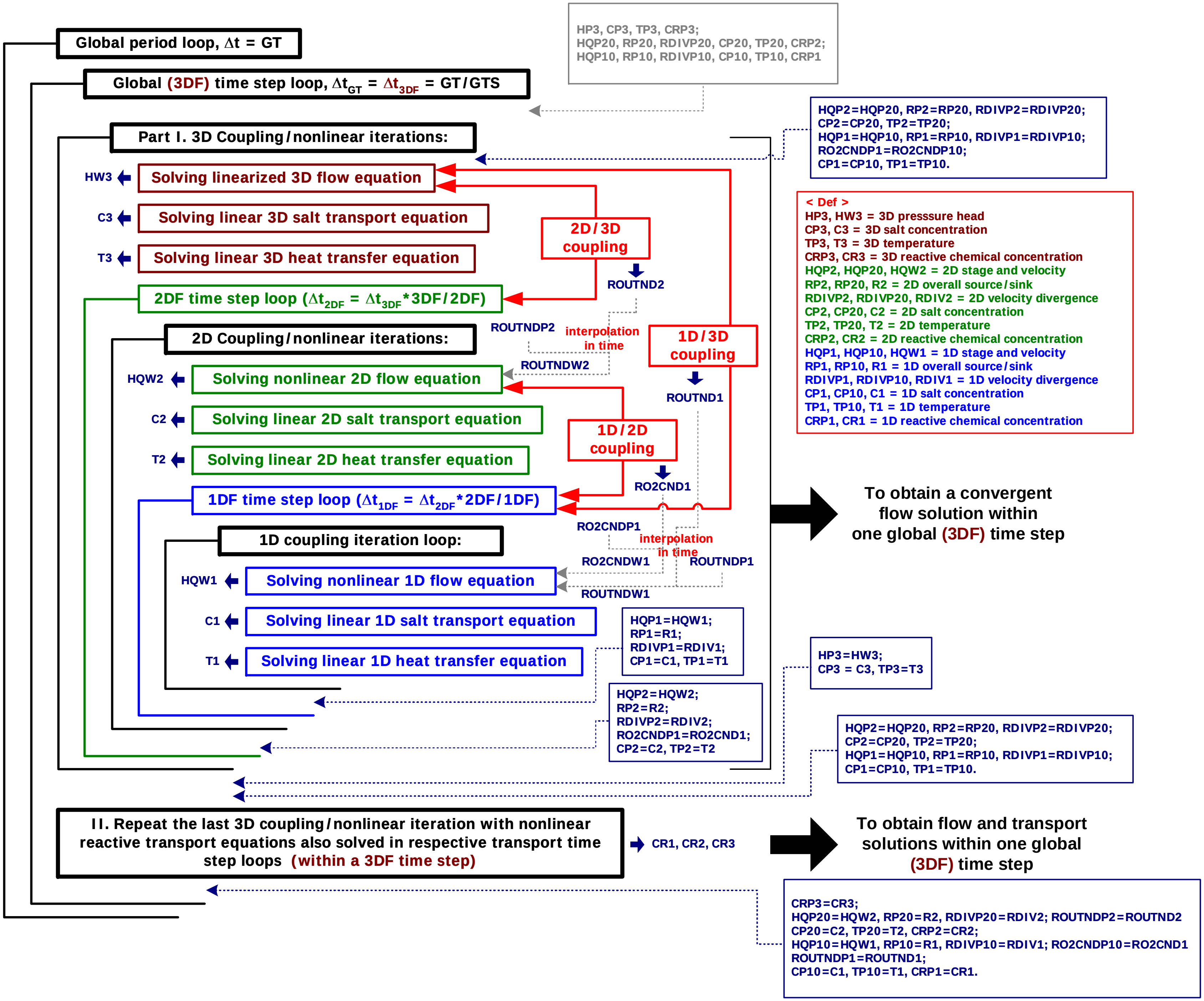

Fig. 3.9-8. Computation Structure of WASH123D for Coupled 1D/2D/3D Simulations