ERDC TN-SWWRP-10-4
August 2010

Computing Locally-Mass-Conservative Fluxes from Multi-dimensional Finite Element Flow Simulations

by Jing-Ru C. Cheng, Hwai-Ping Cheng, Matthew W. Farthing, and Christopher E. Kees

PURPOSE: This technical note details how conservative normal fluxes through element edges and faces are computed in two- (2-D) and three-dimensional (3-D) spaces in a flux-calculation post-processor developed in the System-Wide Water Resources Program (SWWRP).

BACKGROUND: Conserving local mass in the finite volume (FV) sense, where the sum of face fluxes equal to the rate of change of storage within each element/cell, is essential in computing water flow and contaminant transport. Although the continuous Galerkin finite element (FE) method does not yield locally conservative flux approximation directly, Berger and Howington showed that, by remaining consistent with the discrete approximation given by the FE statement, the resulting flux estimates will preserve mass balance [1]. Passing conservative water flux through each element edge/face from flow models to transport models is critical for accurate simulation and analysis. At the U.S. Army Engineer Research and Development Center (ERDC), most water flow models employ the FE method, while many contaminant transport models use the FV method. The computation of locally-conservative water flux through each elemental edge/face has thus become necessary for passing FE-based water flux to FV-based contaminant transport models. This technical note describes how, from multi-dimensional FE-based flow simulations, we computed locally-mass-conservative fluxes to hand-off to FV-based models and to eliminate apparent flux jumps on element boundaries for particle tracking. The computed conservative flux can also be used to set boundary conditions for inset models when desired.

METHODOLOGY: We employed the two flux post-processing techniques focused in [2]. For convenience, we refer to the first technique as the local approach, and the second one the global approach in this technical note. Both were originally developed by Larson and Niklasson [3]. Larson and Niklasson's global approach is mathematically equivalent to Sun and Wheeler's global approach [4]. For both the local and the global approaches, the required input data include

(1) the element residuals associated with the respective computed solution, meaning they are computed by constructing matrix equations at the element level with flux-type boundary conditions incorporated (note that each element has residuals matching its element nodes to satisfy local conservation);

(2) boundary face information, including
   a. an indicator to show whether this boundary face is specified with a flux-type boundary condition:
      - a no-flow boundary face is specified with a zero-flux boundary condition,

- a flow-through (open) boundary face specified with a flux-type boundary condition will have the normal flux match the specified value,
- a flow-through (open) boundary face without a flux-type boundary condition applied will have the normal flux computed.

b. the specified normal flux for this boundary face (set to zero if the normal flux through this boundary face is not to be computed)

(3) the estimated fluxes associated with element faces;

(4) the element connectivity information;

(5) nodal coordinates.

From the input data (3) – (5) above, we can derive the following information for conservative normal flux computation:

(1) node-element connectivity,

(2) edge-element (2-D) or face-element (3-D) connectivity,

(3) edge-node (2-D) or face-node (3-D) connectivity,

(4) edge length (2-D) or face area (3-D).

It is noted that a no-flow boundary face, by definition, is associated with a zero-flux boundary condition, and thus the normal flux through this boundary face is zero. On the other hand, a boundary face is flow-through if it is not associated with a no-flow boundary condition. A flow-through boundary face can be assigned a non-zero flux-type boundary condition, or its associated boundary flux is computed based on the solution of the continuity equation. This concept is applied when we compute conservative fluxes through element faces associated with boundary nodes.

Figure 1 depicts the flow chart for computing conservative normal fluxes in the post-processor: reading input data in Step 1, followed by generating connectivity information, face length/area, and boundary flux indicator in Step 2, then the computation of an artificial, element-based quantity used for normal flux correction in Step 3, and finally in Step 4 the computation of conservative normal fluxes through all element faces. In the following, we discuss the computation involved in Steps 3 and 4 with the local and the global approaches.

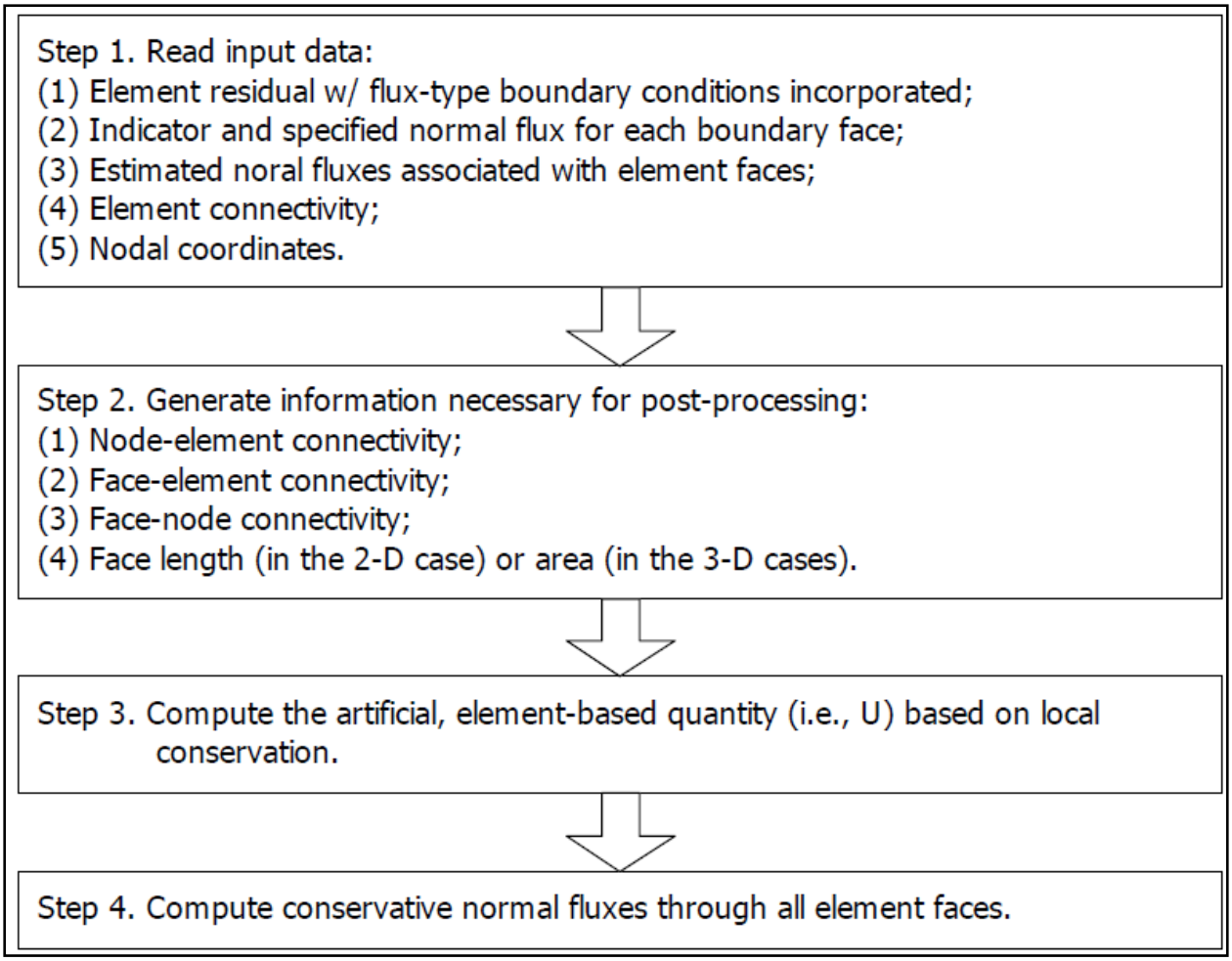

Figure 1. Flow chart for computing conservative normal flux. "Face" is used in the chart to represent 2-D edge or 3-D face.

The Local Approach. In the local approach, the conservative sub-edge (in 2-D) or sub-face (in 3-D)[1] normal fluxes associated with element faces around each node are computed on a node-by-node basis. The conservative normal flux of an element edge/face is computed then by combining all of the associated sub-edge or sub-face normal fluxes.

Scenario 1: For element edges associated with an internal node. As shown in Figure 2 below, there are 6 elements connected at Node 1, and there are six sub-edge flows satisfying the discretized governing equation of continuity, e.g., Richards' equation. In Figure 2(b), the three residuals, each associated with a node, of each element are marked. These residuals represent the net flow entering or leaving the element, and they can be computed by substituting the numerical

[1] A sub-edge or a sub-face of a node is an elemental edge/face containing the node. A sub-edge has a length equal to half of the element edge length, while a sub-face has an area equal to the element face area divided by the number of face nodes. It is called a sub-edge or a sub-face because the conservative normal flux computed for a sub-edge or sub-face is associated with only a portion of edge length or face area. For example, in Figure 2, there are six element edges around Node 1: $I_{1\text{-}2}$, $I_{1\text{-}3}$, $I_{1\text{-}4}$, $I_{1\text{-}5}$, $I_{1\text{-}6}$, $I_{1\text{-}7}$. While each of these six edges is a sub-edge of Node 1, each sub-edge has a length only half of the corresponding element edge length.

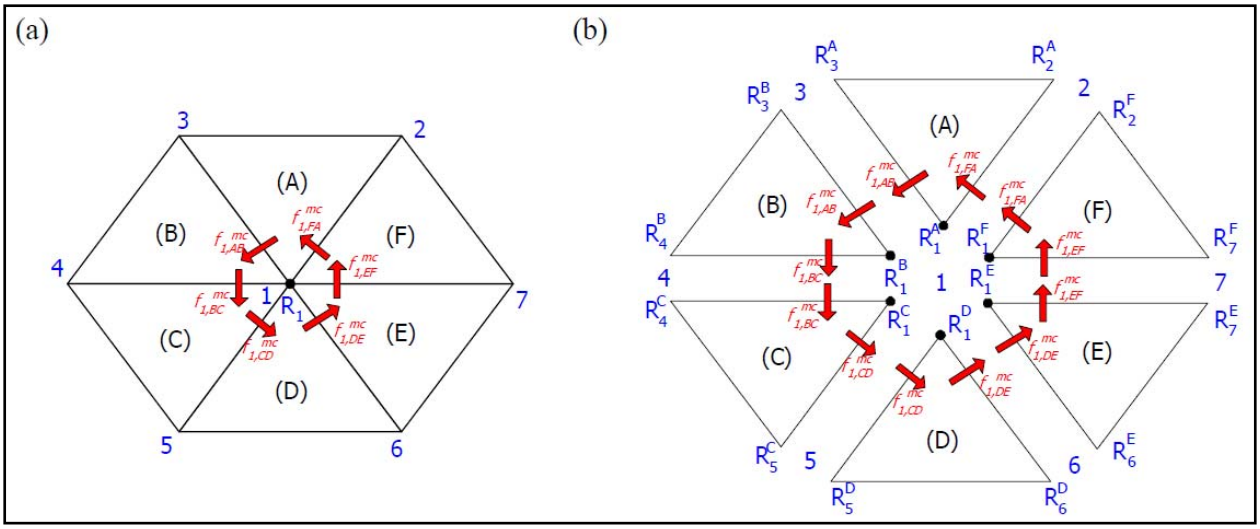

Figure 2. The six triangular elements associated with an internal node (i.e., Node 1) and the six conservative sub-edge flows around Node 1: (a) elements are connected, (b) elements are separated.

solution of the continuity equation back into the element matrix equation constructed in the FE integration. For example, if we solve the Richards' equation for subsurface flow using the Galerkin FE method, the element residual at local node j can be defined as

$$
\begin{aligned}
R_j^e = &-\left[\int_{\Omega_e} N_i^e F N_j^e d\Omega_e\right]\frac{dh_j}{dt} - \left[\int_{\Omega_e}\left(\nabla N_i^e\right)\cdot\mathbf{K}\cdot\left(\nabla N_j^e\right)d\Omega_e\right]h_j + \int_{\Omega_e} N_i^e q\, d\Omega_e \\
&- \int_{\Omega_e}\left(\nabla N_i^e\right)\cdot\mathbf{K}\cdot\nabla z d\Omega_e \qquad (1)\\
= &-\int_{\Gamma_e}\mathbf{n}\cdot\mathbf{K}\cdot\left(\nabla h + \nabla z\right)N_i^e d\Gamma_e
\end{aligned}
$$

where

R_j^e = the element residual at local node j

Ω_e = the domain of element e

N_i^e = the i-th local base function of element e

F = water capacity associated with element e

h_j = pressure head at local node j

∇ = the del operator

$\mathbf{K}$ = hydraulic conductivity tensor associated with element e

q = source/sink in element e

h = pressure head

z = gravity head

Γ_e = the boundary of element e

Here, R_j^e represents the net outgoing flow associated with local node j when element e is considered. Therefore in Figure 2(b), R_1^A, R_2^A, and R_3^A are the three element residuals of Element (A); R_1^B, R_3^B, and R_4^B are the three element residuals of Element (B), and so on. Because Node 1 is an internal node and there are no sources or sinks, the sum of all the six element residuals associated with Node 1 is zero, i.e.,

$$R_1 = R_1^A + R_1^B + R_1^C + R_1^D + R_1^E + R_1^F = 0 \tag{2}$$

where

R_i = total element residual associated with Node i

R_i^E = element residual associated with Node i coming from element (E)

and, R_1^A can be expressed as

$$R_1^A = f_{1,AB}^{mc} - f_{1,FA}^{mc} = V_{1,AB}^{mc} \cdot \frac{l_{AB}}{2} - V_{1,FA}^{mc} \cdot \frac{l_{FA}}{2} \tag{3}$$

where

$f_{i,E_1E_2}^{mc}$ = normal sub-edge flow from Element (E_1) to Element (E_2), which conserves mass locally around Node i

$V_{i,E_1E_2}^{mc}$ = normal sub-edge flux from Element (E_1) to Element (E_2), which conserves mass locally around Node i

$l_{E_1E_2}$ = edge length associated with the interface between Elements (E_1) and (E_2)

If $V_{1,AB}$ represents a given estimated normal sub-edge flux, we can relate $V_{1,AB}^{mc}$ to $V_{1,AB}$ with a correction term as

$$V_{1,AB}^{mc} = V_{1,AB} + \Delta V_{1,AB} = V_{1,AB} + \left(U_{1,B} - U_{1,A}\right) \tag{4}$$

where $U_{1,A}$ and $U_{1,B}$ are artificial, element-based quantities introduced to help calculate the correction of normal sub-edge flux.

Therefore, Equation 3 can be written as:

$$R_1^A = \left[V_{1,AB} + \left(U_{1,B} - U_{1,A}\right)\right] \cdot \frac{l_{AB}}{2} - \left[V_{1,FA} + \left(U_{1,A} - U_{1,F}\right)\right] \cdot \frac{l_{FA}}{2} \tag{5}$$

Likewise, R_1^B, R_1^C, R_1^D, R_1^E, R_1^F can be expressed as

$$R_1^B = \left[V_{1,BC} + \left(U_{1,C} - U_{1,B}\right)\right] \cdot \frac{l_{BC}}{2} - \left[V_{1,AB} + \left(U_{1,B} - U_{1,A}\right)\right] \cdot \frac{l_{AB}}{2} \tag{6}$$

$$R_1^C = \left[V_{1,CD} + \left(U_{1,D} - U_{1,C}\right)\right] \cdot \frac{l_{CD}}{2} - \left[V_{1,BC} + \left(U_{1,C} - U_{1,B}\right)\right] \cdot \frac{l_{BC}}{2} \tag{7}$$

$$R_1^D = \left[V_{1,DE} + \left(U_{1,E} - U_{1,D}\right)\right] \cdot \frac{l_{DE}}{2} - \left[V_{1,CD} + \left(U_{1,D} - U_{1,C}\right)\right] \cdot \frac{l_{CD}}{2} \tag{8}$$

$$R_1^E = \left[V_{1,EF} + \left(U_{1,F} - U_{1,E}\right)\right] \cdot \frac{l_{EF}}{2} - \left[V_{1,DE} + \left(U_{1,E} - U_{1,D}\right)\right] \cdot \frac{l_{DE}}{2} \tag{9}$$

$$R_1^F = \left[V_{1,FA} + \left(U_{1,A} - U_{1,F}\right)\right] \cdot \frac{l_{FA}}{2} - \left[V_{1,EF} + \left(U_{1,F} - U_{1,E}\right)\right] \cdot \frac{l_{EF}}{2} \tag{10}$$

It must be noted that every equation from Equation 5 through Equation 10 can be represented by the other five equations, i.e., there are only five independent equations. To solve the six unknowns, we enforce $U_{1,A} = 0$ and use it to replace one of the six equations, say Equation 5. This will not affect the resulting flux correction mathematically because Equations 6 through Equation 10 correspond to a Neumann problem, as mentioned in Kees et al. [2]. It is because the flux correction is the jump of two adjacent *U's*, rather than the *U* values. Setting $U_{1,A}$ to any value will still produce the same jump values as long as the element residual equations are solved accurately.

After solving Equation 6 through Equation 10, the conservative normal sub-edge fluxes can be computed with

$$V_{1,AB}^{mc} = V_{1,AB} + \left(U_{1,B} - 0\right) \tag{11a}$$

$$V_{1,BC}^{mc} = V_{1,BC} + \left(U_{1,C} - U_{1,B}\right) \tag{11b}$$

$$V_{1,CD}^{mc} = V_{1,CD} + \left(U_{1,D} - U_{1,C}\right) \tag{11c}$$

$$V_{1,DE}^{mc} = V_{1,DE} + \left(U_{1,E} - U_{1,D}\right) \tag{11d}$$

$$V_{1,EF}^{mc} = V_{1,EF} + \left(U_{1,F} - U_{1,E}\right) \tag{11e}$$

$$V_{1,FA}^{mc} = V_{1,FA} + \left(0 - U_{1,F}\right) \tag{11f}$$

Scenario 2: For element edges associated with a boundary node. A boundary edge can be of either flow-through or no-flow. Each flow-through boundary edge may or may not be assigned a specified flux-type boundary condition in numerical simulation. When it is assigned with a specified flux-type boundary condition, its normal flux must be identical to the specified flux. If it is not specified with a flux-type boundary condition, its normal flux can be computed based on the solution of the continuity equation. A no-flow boundary edge, on the other hand, is associated with a zero normal flux by definition. In other words, it is specified with a zero-flux boundary

condition even though in most numerical models a boundary edge is usually defaulted to no-flow type when it is not specified with any type of boundary condition.

As shown in Figure 3, there are three elements around Node 1, which is a boundary node, and Edges 1-2 and 1-5 are the two boundary edges adjacent to Node 1. In this case, there are four normal sub-edge fluxes to be computed to satisfy the discretized continuity equation.

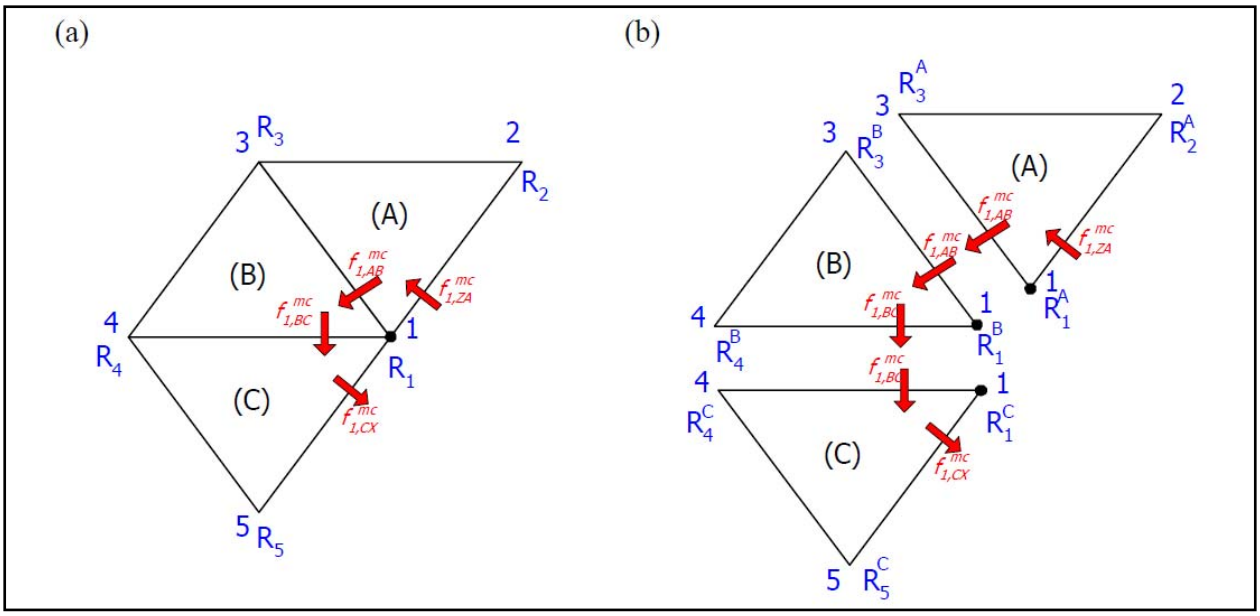

Figure 3. The three triangular elements associated with a boundary node (i.e., Node 1) and the four conservative sub-edge flows around Node 1: (a) elements are connected, (b) elements are separated; Among the four sub-edge flows, two are associated with boundary edges.

Without the flux-type boundary conditions applied, the three element residual equations associated with Node 1 are:

$$\begin{aligned} R_1^A &= f_{1,AB}^{mc} - f_{1,ZA}^{mc} = V_{1,AB}^{mc} \cdot \frac{l_{AB}}{2} - V_{1,ZA}^{mc} \cdot \frac{l_{ZA}}{2} \\ &= \left[V_{1,AB} + \left(U_{1,B} - U_{1,A}\right)\right] \cdot \frac{l_{AB}}{2} - \left[V_{1,ZA} + \left(U_{1,A} - U_{1,Z}\right)\right] \cdot \frac{l_{ZA}}{2} \end{aligned} \tag{12}$$

$$\begin{aligned} R_1^B &= f_{1,BC}^{mc} - f_{1,AB}^{mc} = V_{1,BC}^{mc} \cdot \frac{l_{BC}}{2} - V_{1,AB}^{mc} \cdot \frac{l_{AB}}{2} \\ &= \left[V_{1,BC} + \left(U_{1,C} - U_{1,B}\right)\right] \cdot \frac{l_{BC}}{2} - \left[V_{1,AB} + \left(U_{1,B} - U_{1,A}\right)\right] \cdot \frac{l_{AB}}{2} \end{aligned} \tag{13}$$

$$\begin{aligned} R_1^C &= f_{1,CX}^{mc} - f_{1,BC}^{mc} = V_{1,CX}^{mc} \cdot \frac{l_{CX}}{2} - V_{1,BC}^{mc} \cdot \frac{l_{BC}}{2} \\ &= \left[V_{1,CX} + \left(U_{1,X} - U_{1,C}\right)\right] \cdot \frac{l_{CX}}{2} - \left[V_{1,BC} + \left(U_{1,C} - U_{1,B}\right)\right] \cdot \frac{l_{BC}}{2} \end{aligned} \tag{14}$$

where $f_{1,ZA}^{mc}$ and $f_{1,CX}^{mc}$ are the sub-edge flow through Boundary Edges 1-2 and 1-5, respectively; l_{ZA} and l_{CX} are the edge lengths associated with Boundary Edges 1-2 and 1-5, respectively; $U_{1,Z}$ and $U_{1,X}$ are the U values associated imaginary elements outside of the computational domain.

Without knowing the relationship between $U_{1,X}$ and $U_{1,Z}$, we can simply assume and enforce $U_{1,X} = U_{1,Z} = C$ in Equation 12 and Equation 14 to help close the computational system, where C is a constant. The correction of normal sub-edge flux (i.e., the jump of U between connected elements) is uniquely determined as long as $U_{1,X}$ and $U_{1,Z}$ are set to the same value, which is reasonable because they represent outside quantities around Node 1. By setting C = 0, Equations 12 through 14 thus become

$$R_1^A = f_{1,AB}^{mc} - f_{1,ZA}^{mc} = \left[V_{1,AB} + \left(U_{1,B} - U_{1,A}\right)\right] \cdot \frac{l_{AB}}{2} - \left[V_{1,ZA} + \left(U_{1,A} - 0\right)\right] \cdot \frac{l_{ZA}}{2} \tag{15}$$

$$R_1^B = f_{1,BC}^{mc} - f_{1,AB}^{mc} = \left[V_{1,BC} + \left(U_{1,C} - U_{1,B}\right)\right] \cdot \frac{l_{BC}}{2} - \left[V_{1,AB} + \left(U_{1,B} - U_{1,A}\right)\right] \cdot \frac{l_{AB}}{2} \tag{16}$$

$$R_1^C = f_{1,CX}^{mc} - f_{1,BC}^{mc} = \left[V_{1,CX} + \left(0 - U_{1,C}\right)\right] \cdot \frac{l_{CX}}{2} - \left[V_{1,BC} + \left(U_{1,C} - U_{1,B}\right)\right] \cdot \frac{l_{BC}}{2} \tag{17}$$

In solving Equation 15 through Equation 17, we consider three possible situations concerning the specified flux-type boundary condition: (1) when all boundary edges are assigned; (2) when some, but not all, boundary edges are assigned; (3) when no boundary edge is assigned.

Situation 1. All boundary edges are assigned flux-type boundary conditions: When flux-type boundary conditions have been taken into account in computing the element residuals, $f_{1,ZA}^{mc}$ and $f_{1,CX}^{mc}$ will no longer appear in the element residual equations. If q_{ZA} and q_{CX} represent the outward boundary flux through Edges 1-2 and 1-5, respectively, Equations 15 through 17 become:

$$R_1^{A,bf} = R_1^A - \frac{q_{ZA} \cdot l_{ZA}}{2} = \left[V_{1,AB} + \left(U_{1,B} - U_{1,A}\right)\right] \cdot \frac{l_{AB}}{2} \tag{18}$$

$$R_1^{B,bf} = R_1^B = \left[V_{1,BC} + \left(U_{1,C} - U_{1,B}\right)\right] \cdot \frac{l_{BC}}{2} - \left[V_{1,AB} + \left(U_{1,B} - U_{1,A}\right)\right] \cdot \frac{l_{AB}}{2} \tag{19}$$

$$R_1^{C,bf} = R_1^C - \frac{q_{CX} \cdot l_{CX}}{2} = -\left[V_{1,BC} + \left(U_{1,C} - U_{1,B}\right)\right] \cdot \frac{l_{BC}}{2} \tag{20}$$

where $R_1^{A,bf}$, $R_1^{B,bf}$, and $R_1^{C,bf}$ represent element residuals with flux-type boundary conditions taken into account.

Equation 18 is linearly dependent on Equation 19 and Equation 20. This is again a Neumann problem, and we can enforce $U_{1,A} = 0$ and solve Equation 19 and Equation 20 for $U_{1,B}$ and $U_{1,C}$. The normal sub-edge fluxes can be computed then with

$$V_{1,ZA}^{mc} = -q_{ZA} \tag{21a}$$

$$V_{1,AB}^{mc} = V_{1,AB} + \left(U_{1,B}\right) \tag{21b}$$

$$V_{1,BC}^{mc} = V_{1,BC} + \left(U_{1,C} - U_{1,B}\right) \tag{21c}$$

$$V_{1,CX}^{mc} = q_{CX} \tag{21d}$$

Situation 2. Some, but not all, boundary edges are assigned flux-type boundary conditions: In the two-dimensional case here, we consider when only one of the two connected boundary edges is assigned a flux-type boundary condition. Suppose the outward boundary flux through Edge 1-2 is q_{ZA}, Equation 15 through Equation 17 become:

$$R_1^{A,bf} = R_1^A - \frac{q_{ZA} \cdot l_{ZA}}{2} = \left[V_{1,AB} + \left(U_{1,B} - U_{1,A}\right)\right] \cdot \frac{l_{AB}}{2} \tag{22}$$

$$R_1^{B,bf} = R_1^B = \left[V_{1,BC} + \left(U_{1,C} - U_{1,B}\right)\right] \cdot \frac{l_{BC}}{2} - \left[V_{1,AB} + \left(U_{1,B} - U_{1,A}\right)\right] \cdot \frac{l_{AB}}{2} \tag{23}$$

$$R_1^{C,bf} = R_1^C = \left[V_{1,CX} + \left(0 - U_{1,C}\right)\right] \cdot \frac{l_{CX}}{2} - \left[V_{1,BC} + \left(U_{1,C} - U_{1,B}\right)\right] \cdot \frac{l_{BC}}{2} \tag{24}$$

We thus solve Equation 22 through Equation 24 for $U_{1,A}$, $U_{1,B}$, and $U_{1,C}$. Then the normal sub-edge fluxes can be computed with

$$V_{1,ZA}^{mc} = -q_{ZA} \tag{25a}$$

$$V_{1,AB}^{mc} = V_{1,AB} + \left(U_{1,B} - U_{1,A}\right) \tag{25b}$$

$$V_{1,BC}^{mc} = V_{1,BC} + \left(U_{1,C} - U_{1,B}\right) \tag{25c}$$

$$V_{1,CX}^{mc} = V_{1,CX} + \left(0 - U_{1,C}\right) \tag{25d}$$

Situation 3. None of the boundary edges is assigned a flux-type boundary condition: In this case, we solve Equation 15 through Equation 17 for $U_{1,A}$, $U_{1,B}$, and $U_{1,C}$, and compute the normal sub-edge fluxes by using

$$V_{1,ZA}^{mc} = V_{1,ZA} + \left(U_{1,A} - 0\right) \tag{26a}$$

$$V_{1,AB}^{mc} = V_{1,AB} + \left(U_{1,B} - U_{1,A}\right) \tag{26b}$$

$$V_{1,BC}^{mc} = V_{1,BC} + \left(U_{1,C} - U_{1,B}\right) \tag{26c}$$

$$V_{1,CX}^{mc} = V_{1,CX} + \left(0 - U_{1,C}\right) \tag{26d}$$

Summary of sub-edge flux computation. Suppose there are M elements connected at a node, the following can be drawn from the discussion above in the process of constructing the element residual equations around the node:

(1) Use element residuals from the FE integration to construct the M element residual equations, one for each element.

(2) If the node is an internal node, set one *U* value to zero and solve the others.

(3) If the node is a boundary node, set the "outside" *U*'s to zero (e.g., $U_{1,X} = U_{1,Z} = 0$).

(4) If the node is a boundary node and all the connected boundary edges are assigned flux-type boundary conditions, set one *U* value to zero and solve the others.

The constructed element residual equations are then solved with a full-pivoting direct solver to minimize possible numerical error. The conservative sub-edge fluxes are computed using the computed *U*'s as well as the specified boundary fluxes.

Compute edge fluxes based on sub-edge fluxes. The conservative normal flux through an element edge can then be computed by taking arithmetic average of its two corresponding sub-edge fluxes. In Figure 2, for example, the two conservative normal sub-edge fluxes of Edge 1-2 are $V_{1,AB}^{mc}$ and $V_{2,AB}^{mc}$. The conservative normal flux of the edge (i.e., V_{AB}^{mc}) is computed as

$$V_{AB}^{mc} = \frac{V_{1,AB}^{mc} + V_{2,AB}^{mc}}{2} \tag{27}$$

Extension to three-dimensional space. The extension of the local approach from 2- to 3-dimensional space is straightforward. For easy visualization, we use hexahedral elements for demonstration. By using the summary of sub-face flux computation above, we construct the 3-D element residual equations for the corresponding scenario as follows.

Scenario 1. Around an internal node: As shown in Figure 4, Node 14 is an internal node connected to eight hexahedral elements and associated with 12 sub-face fluxes. The eight element residual equations associated with Node 14 are

$$U_{14,A} = 0 \tag{28}$$

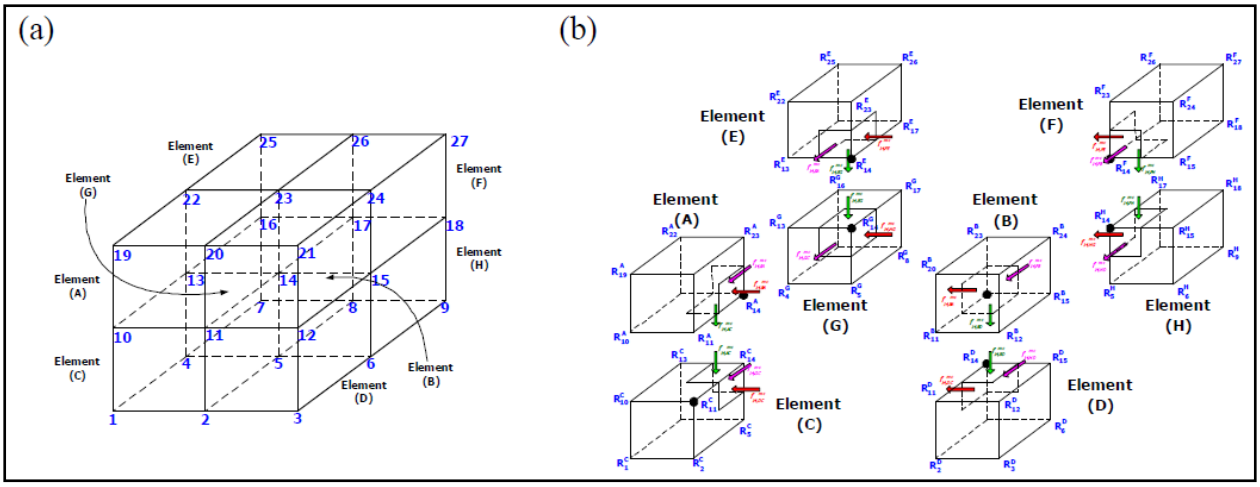

Figure 4. The eight hexahedral elements associated with an internal node (i.e., Node 14) and the 12 conservative sub-face flows around Node 14: (a) elements are connected, (b) elements are separated.

$$
\begin{aligned}
R_{14}^{B} &= f_{14,BA}^{mc} - f_{14,FB}^{mc} + f_{14,BD}^{mc} \\
&= \left[V_{14,BA} + \left(U_{14,A} - U_{14,B}\right)\right] \cdot \frac{A_{BA}}{4} - \left[V_{14,FB} + \left(U_{14,B} - U_{14,F}\right)\right] \cdot \frac{A_{FB}}{4} \\
&\quad + \left[V_{14,BD} + \left(U_{14,D} - U_{14,B}\right)\right] \cdot \frac{A_{BD}}{4}
\end{aligned} \tag{29}
$$

$$
\begin{aligned}
R_{14}^{C} &= -f_{14,DC}^{mc} - f_{14,GC}^{mc} - f_{14,AC}^{mc} \\
&= -\left[V_{14,DC} + \left(U_{14,C} - U_{14,D}\right)\right] \cdot \frac{A_{DC}}{4} - \left[V_{14,GC} + \left(U_{14,C} - U_{14,G}\right)\right] \cdot \frac{A_{GC}}{4} \\
&\quad - \left[V_{14,AC} + \left(U_{14,C} - U_{14,A}\right)\right] \cdot \frac{A_{AC}}{4}
\end{aligned} \tag{30}
$$

$$
\begin{aligned}
R_{14}^{D} &= f_{14,DC}^{mc} - f_{14,HD}^{mc} - f_{14,BD}^{mc} \\
&= \left[V_{14,DC} + \left(U_{14,C} - U_{14,D}\right)\right] \cdot \frac{A_{DC}}{4} - \left[V_{14,HD} + \left(U_{14,D} - U_{14,H}\right)\right] \cdot \frac{A_{HD}}{4} \\
&\quad - \left[V_{14,BD} + \left(U_{14,D} - U_{14,B}\right)\right] \cdot \frac{A_{BD}}{4}
\end{aligned} \tag{31}
$$

$$
\begin{aligned}
R_{14}^{E} &= -f_{14,FE}^{mc} + f_{14,EA}^{mc} + f_{14,EG}^{mc} \\
&= -\left[V_{14,FE} + \left(U_{14,E} - U_{14,F}\right)\right] \cdot \frac{A_{FE}}{4} + \left[V_{14,EA} + \left(U_{14,A} - U_{14,E}\right)\right] \cdot \frac{A_{EA}}{4} \\
&\quad + \left[V_{14,EG} + \left(U_{14,G} - U_{14,E}\right)\right] \cdot \frac{A_{EG}}{4}
\end{aligned} \tag{32}
$$

$$
\begin{aligned}
R_{14}^{F} &= f_{14,FE}^{mc} + f_{14,FB}^{mc} + f_{14,FH}^{mc} \\
&= \left[V_{14,FE} + \left(U_{14,E} - U_{14,F}\right)\right] \cdot \frac{A_{FE}}{4} + \left[V_{14,FB} + \left(U_{14,B} - U_{14,F}\right)\right] \cdot \frac{A_{FB}}{4} \\
&\quad + \left[V_{14,FH} + \left(U_{14,H} - U_{14,F}\right)\right] \cdot \frac{A_{FH}}{4}
\end{aligned} \tag{33}
$$

$$
\begin{aligned}
R_{14}^{G} &= -f_{14,HG}^{mc} + f_{14,GC}^{mc} - f_{14,EG}^{mc} \\
&= -\left[V_{14,HG} + \left(U_{14,G} - U_{14,H}\right)\right] \cdot \frac{A_{HG}}{4} + \left[V_{14,GC} + \left(U_{14,C} - U_{14,G}\right)\right] \cdot \frac{A_{GC}}{4} \\
&\quad - \left[V_{14,EG} + \left(U_{14,G} - U_{14,E}\right)\right] \cdot \frac{A_{EG}}{4}
\end{aligned} \tag{34}
$$

$$
\begin{aligned}
R_{14}^{H} &= f_{14,HG}^{mc} + f_{14,HD}^{mc} - f_{14,FH}^{mc} \\
&= \left[V_{14,HG} + \left(U_{14,G} - U_{14,H}\right)\right] \cdot \frac{A_{HG}}{4} + \left[V_{14,HD} + \left(U_{14,D} - U_{14,H}\right)\right] \cdot \frac{A_{HD}}{4} \\
&\quad - \left[V_{14,FH} + \left(U_{14,H} - U_{14,F}\right)\right] \cdot \frac{A_{FH}}{4}
\end{aligned} \tag{35}
$$

where A_{BA} denotes the face area associated with the interface between Elements (B) and (A); and R is element residual, V is estimated normal sub-face flux, U is sub-element-based quantity employed for flux correction, as defined previously. It is noted that the sub-face area is equal to the face area divided by the number of face nodes, e.g., it is 4 for quadrilateral and 3 for triangular faces.

Scenario 2. Around a boundary node: In Figure 5, Node 8 is a boundary node at which four hexahedral elements are connected. It is associated with eight sub-face fluxes, where four of them are associated with boundary faces. By setting $U_{8,W} = U_{8,X} = U_{8,Y} = U_{8,Z} = 0$, the four element residual equations associated with Node 8 are

$$
\begin{aligned}
R_{8}^{A} &= -f_{8,BA}^{mc} + f_{8,AW}^{mc} + f_{8,AC}^{mc} \\
&= -\left[V_{8,BA} + \left(U_{8,A} - U_{8,B}\right)\right] \cdot \frac{A_{BA}}{4} + \left[V_{8,AW} + \left(0 - U_{8,A}\right)\right] \cdot \frac{A_{AW}}{4} + \left[V_{8,AC} + \left(U_{8,C} - U_{8,A}\right)\right] \cdot \frac{A_{AC}}{4}
\end{aligned} \tag{36}
$$

$$
\begin{aligned}
R_{8}^{B} &= f_{8,BA}^{mc} + f_{8,BX}^{mc} + f_{8,BD}^{mc} \\
&= \left[V_{8,BA} + \left(U_{8,A} - U_{8,B}\right)\right] \cdot \frac{A_{BA}}{4} + \left[V_{8,BX} + \left(0 - U_{8,B}\right)\right] \cdot \frac{A_{BX}}{4} + \left[V_{8,BD} + \left(U_{8,D} - U_{8,B}\right)\right] \cdot \frac{A_{BD}}{4}
\end{aligned} \tag{37}
$$

$$
\begin{aligned}
R_{8}^{C} &= -f_{8,DC}^{mc} + f_{8,CY}^{mc} - f_{8,AC}^{mc} \\
&= -\left[V_{8,DC} + \left(U_{8,C} - U_{8,D}\right)\right] \cdot \frac{A_{DC}}{4} + \left[V_{8,CY} + \left(0 - U_{8,C}\right)\right] \cdot \frac{A_{CY}}{4} - \left[V_{8,AC} + \left(U_{8,C} - U_{8,A}\right)\right] \cdot \frac{A_{AC}}{4}
\end{aligned} \tag{38}
$$

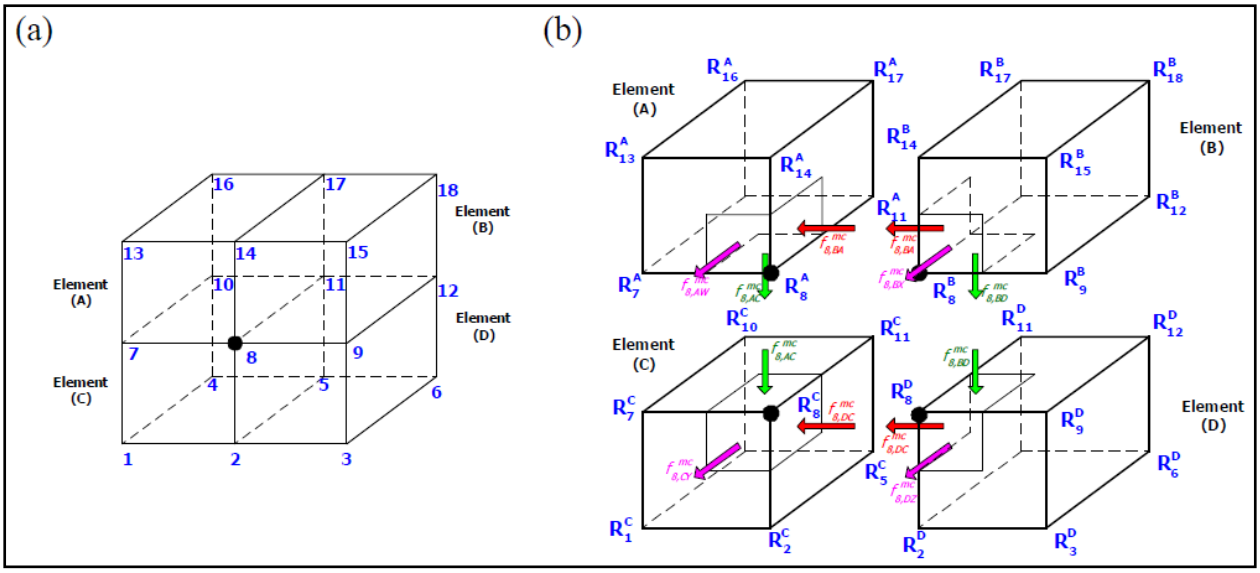

Figure 5. The four hexahedral elements associated with a boundary node (i.e., Node 8) and the eight conservative sub-face flows around Node 8: (a) elements are connected, (b) elements are separated.

$$
\begin{aligned}
R_8^D &= f_{8,DC}^{mc} + f_{8,DZ}^{mc} - f_{8,BD}^{mc} \\
&= \left[V_{8,DC} + \left(U_{8,C} - U_{8,D}\right)\right] \cdot \frac{A_{DC}}{4} + \left[V_{8,DZ} + \left(0 - U_{8,D}\right)\right] \cdot \frac{A_{DZ}}{4} - \left[V_{8,BD} + \left(U_{8,D} - U_{8,B}\right)\right] \cdot \frac{A_{BD}}{4}
\end{aligned}
\tag{39}
$$

Because the specified boundary fluxes are already accounted for in computing element residuals, Equation 36 through Equation 39 will be modified if they involve flux-type boundary faces. For example, if Boundary Faces 7-8-14-13 and 8-9-15-14 are assigned flux-type boundary conditions, Equation 36 and Equation 37 become

$$
\begin{aligned}
R_8^{A,bf} &= R_8^A - f_{8,AW}^{mc} = -f_{8,BA}^{mc} + f_{8,AC}^{mc} \\
&= -\left[V_{8,BA} + \left(U_{8,A} - U_{8,B}\right)\right] \cdot \frac{A_{BA}}{4} + \left[V_{8,AC} + \left(U_{8,C} - U_{8,A}\right)\right] \cdot \frac{A_{AC}}{4}
\end{aligned}
\tag{40}
$$

$$
\begin{aligned}
R_8^{B,bf} &= R_8^B - f_{8,BX}^{mc} = f_{8,BA}^{mc} + f_{8,BD}^{mc} \\
&= \left[V_{8,BA} + \left(U_{8,A} - U_{8,B}\right)\right] \cdot \frac{A_{BA}}{4} + \left[V_{8,BD} + \left(U_{8,D} - U_{8,B}\right)\right] \cdot \frac{A_{BD}}{4}
\end{aligned}
\tag{41}
$$

In this case, we solve Equation 38 through Equation 41 for $U_{8,A}$, $U_{8,B}$, $U_{8,C}$, $U_{8,D}$.

When all the boundary faces associated with Node 8 are specified with flux-type boundary conditions, Equation 38 and Equation 39 become

$$R_8^{C,bf} = R_8^C - f_{8,CY}^{mc} = -f_{8,DC}^{mc} - f_{8,AC}^{mc}$$
$$= -\left[V_{8,DC} + \left(U_{8,C} - U_{8,D}\right)\right] \cdot \frac{A_{DC}}{4} - \left[V_{8,AC} + \left(U_{8,C} - U_{8,A}\right)\right] \cdot \frac{A_{AC}}{4} \tag{42}$$

$$R_8^{D,bf} = R_8^D - f_{8,DZ}^{mc} = f_{8,DC}^{mc} - f_{8,BD}^{mc}$$
$$= \left[V_{8,DC} + \left(U_{8,C} - U_{8,D}\right)\right] \cdot \frac{A_{DC}}{4} - \left[V_{8,BD} + \left(U_{8,D} - U_{8,B}\right)\right] \cdot \frac{A_{BD}}{4} \tag{43}$$

Now the four element residual equations are Equation 40 through Equation 43. But one of them has to be replaced by setting one U value to zero due to linear dependency among the four equations, as discussed before. If here we set $U_{8,A} = 0$, and use it to replace Equation 40, we are to solve Equation 41 through Equation 43 for $U_{8,B}$, $U_{8,C}$, $U_{8,D}$. The computed U's are then used to correct normal sub-face flux, followed by taking arithmetic average of the conservative sub-face fluxes to obtain conservative normal face flux.

The Global Approach. In the global approach, the conservative normal fluxes are first related to element residuals at the element level to satisfy the local conservation. A global residual equation (in matrix form) is then formed by assembling all element residual equations and solved for all the element-based U's. These U's are used to correct the normal fluxes through element edges/faces for mass conservation.

Scenario 1: For elements whose edges are all internal. As shown in Figure 6 below, Element (A) is an internal element: all of its element edges are internal. Therefore, the following element residual equation exists for Element (A):

$$R^A = R_1^A + R_2^A + R_3^A = f_{AB}^{mc} - f_{CA}^{mc} + f_{AD}^{mc} = V_{AB}^{mc} \cdot l_{AB} - V_{CA}^{mc} \cdot l_{CA} + V_{AD}^{mc} \cdot l_{AD} \tag{44}$$

where

R^E = total element residual of element (E)
R_i^E = element residual associated with Node i at element (E)
$f_{E_1E_2}^{mc}$ = normal edge flow from Element (E_1) to Element (E_2), which conserves mass locally
$V_{E_1E_2}^{mc}$ = normal edge flux from Element (E_1) to Element (E_2), which conserves mass locally
$l_{E_1E_2}$ = edge length associated with the interface between Elements (E_1) and (E_2)

Equation 44 can be written further as

$$R_A = R_{1A} + R_{2A} + R_{3A} = \left[V_{AB} + \left(U_B - U_A\right)\right] \cdot l_{AB} - \left[V_{CA} + \left(U_A - U_C\right)\right] \cdot l_{CA}$$
$$+ \left[V_{AD} + \left(U_D - U_A\right)\right] \cdot l_{AD} \tag{45}$$

where $V_{E_1E_2}$ = given estimated normal edge flux from Element (E_1) to Element (E_2).

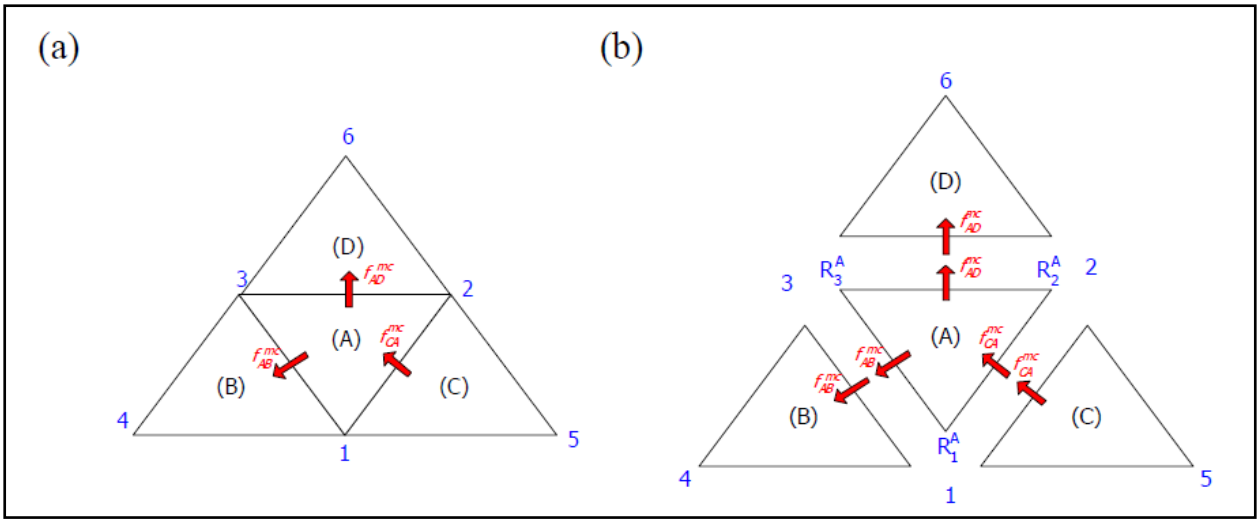

Figure 6. An internal triangular element (i.e., Element (A)) and its three neighbor elements: (a) elements are connected, (b) elements are separated.

Equation 45 is the residual equation for Element (A). Likewise, we can compose the residual equation for each internal element. The conservative fluxes through the three internal edges of element (A) are computed with

$$V_{AB}^{mc} = V_{AB} + \left(U_B - U_A\right) \quad \text{(through Edge 1-3)} \tag{46a}$$

$$V_{CA}^{mc} = V_{CA} + \left(U_A - U_C\right) \quad \text{(through Edge 1-2)} \tag{46b}$$

$$V_{AD}^{mc} = V_{AD} + \left(U_D - U_A\right) \quad \text{(through Edge 2-3)} \tag{46c}$$

Scenario 2: For elements that contain boundary edges. As shown in Figure 7 below, Element (A) has two boundary edges (i.e., Edges 1-2 and 1-3) and one internal edge (Edge 2-3). Without the flux-type boundary conditions applied, the residual equation for Element (A) is:

$$\begin{aligned} R_A &= R_{1A} + R_{2A} + R_{3A} = f_{AB}^{mc} + f_{AZ}^{mc} + f_{AX}^{mc} \\ &= \left[V_{AB} + \left(U_B - U_A\right)\right] \cdot l_{AB} + \left[V_{AZ} + \left(U_Z - U_A\right)\right] \cdot l_{AZ} + \left[V_{AX} + \left(U_X - U_A\right)\right] \cdot l_{AX} \end{aligned} \tag{47}$$

Situation 1. All boundary edges are assigned flux-type boundary conditions: When flux-type boundary conditions have been taken into account in computing the element residuals, f_{AZ}^{mc} and f_{AX}^{mc} will no longer appear in the element residual equations. If q_{AZ} and q_{AX} represent the outward boundary flux through Edges 1-2 and 1-3, respectively, Equation 47 becomes:

$$R_A^{bf} = R_A - q_{AZ} - q_{AX} = f_{AB}^{mc} = \left[V_{AB} + \left(U_B - U_A\right)\right] \cdot l_{AB} \tag{48}$$

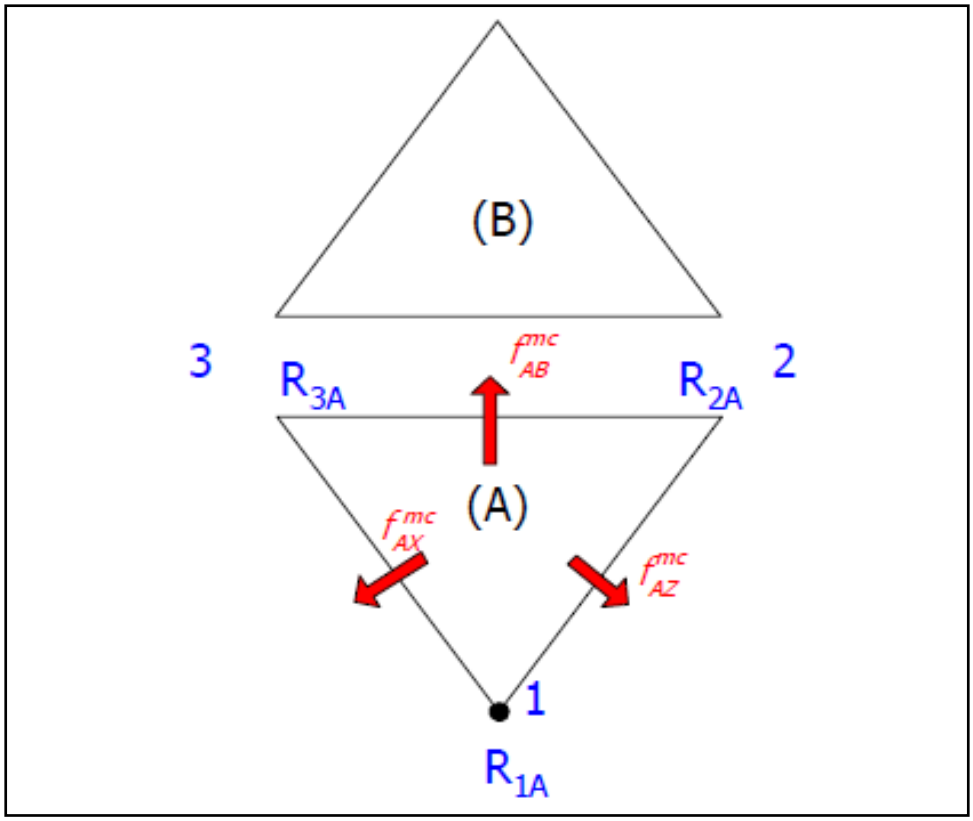

Figure 7. A boundary triangular element (i.e., Element (A)) that has two boundary edges and one internal edge.

The outward normal flux through Edges 1-2, 1-3 and 2-3 are computed by

$$V_{AZ}^{mc} = q_{AZ} \quad \text{(through Edge 1-2)} \tag{49a}$$

$$V_{AX}^{mc} = q_{AX} \quad \text{(through Edge 1-3)} \tag{49b}$$

$$V_{AB}^{mc} = V_{AB} + \left(U_B - U_A\right) \quad \text{(through Edge 2-3)} \tag{49c}$$

Situation 2. Some, but not all, boundary edges are assigned flux-type boundary conditions: In the two-dimensional case here, we consider when only one of the two boundary edges is assigned a flux-type boundary condition. Suppose the outward boundary flux through Edge 1-2 is q_{AZ}, Equation 46 becomes:

$$\begin{aligned} R_A^{bf} = R_A - q_{AZ} &= f_{AB}^{mc} + f_{AC}^{mc} \\ &= \left[V_{AB} + \left(U_B - U_A\right)\right] \cdot l_{AB} + \left[V_{AX} + \left(0 - U_A\right)\right] \cdot l_{AC} \end{aligned} \tag{50}$$

The outward normal fluxes through the three element edges are computed with

$$V_{AZ}^{mc} = q_{AZ} \quad \text{(through Edge 1-2)} \tag{51a}$$

$$V_{AX}^{mc} = V_{AX} + \left(0 - U_A\right) \quad \text{(through Edge 1-3)} \tag{51b}$$

$$V_{AB}^{mc} = V_{AB} + \left(U_B - U_A\right) \quad \text{(through Edge 2-3)} \tag{51c}$$

Situation 3. None of the boundary edges is assigned a flux-type boundary condition: In this case, the outward normal fluxes can be computed by using

$$V_{AZ}^{mc} = V_{AZ} + \left(0 - U_A\right) \quad \text{(through Edge 1-2)} \tag{52a}$$

$$V_{AX}^{mc} = V_{AX} + \left(0 - U_A\right) \quad \text{(through Edge 1-3)} \tag{52b}$$

$$V_{AB}^{mc} = V_{AB} + \left(U_B - U_A\right) \quad \text{(through Edge 2-3)} \tag{52c}$$

Extension to three-dimensional space. The extension of the global approach from two- to three-dimensional space is also straightforward. Again, we use hexahedral elements for demonstration.

Scenario 1: For elements whose faces are all internal. As shown in Figure 8 below, Element (A) is an internal element: all its six element faces are internal. The element residual equation associated with Element (A) is

$$\begin{aligned} R^A &= R_1^A + R_2^A + R_3^A + R_4^A + R_5^A + R_6^A + R_7^A + R_8^A \\ &= f_{AB}^{mc} + f_{AC}^{mc} + f_{AD}^{mc} + f_{AE}^{mc} + f_{AF}^{mc} + f_{AG}^{mc} \\ &= \left[V_{AB} + \left(U_B - U_A\right)\right] \cdot A_{AB} + \left[V_{AC} + \left(U_C - U_A\right)\right] \cdot A_{AC} + \left[V_{AD} + \left(U_D - U_A\right)\right] \cdot A_{AD} \\ &\quad + \left[V_{AE} + \left(U_E - U_A\right)\right] \cdot A_{AE} + \left[V_{AF} + \left(U_F - U_A\right)\right] \cdot A_{AF} + \left[V_{AG} + \left(U_G - U_A\right)\right] \cdot A_{AG} \end{aligned} \tag{53}$$

Scenario 2: For elements that contain boundary faces. As shown in Figure 9 below, Faces 2-3-7-6, 1-2-6-5, and 1-4-3-2 that are associated with Element (A) are boundary faces. Without the flux-type boundary conditions applied, the residual equation for Element 1 is:

$$\begin{aligned} R^A &= R_1^A + R_2^A + R_3^A + R_4^A + R_5^A + R_6^A + R_7^A + R_8^A \\ &= f_{AB}^{mc} + f_{AX}^{mc} + f_{AY}^{mc} + f_{AE}^{mc} + f_{AZ}^{mc} + f_{AG}^{mc} \\ &= \left[V_{AB} + \left(U_B - U_A\right)\right] \cdot A_{AB} + \left[V_{AX} + \left(0 - U_A\right)\right] \cdot A_{AX} + \left[V_{AY} + \left(0 - U_A\right)\right] \cdot A_{AY} \\ &\quad + \left[V_{AE} + \left(U_E - U_A\right)\right] \cdot A_{AE} + \left[V_{AZ} + \left(0 - U_A\right)\right] \cdot A_{AF} + \left[V_{AG} + \left(U_G - U_A\right)\right] \cdot A_{AG} \end{aligned} \tag{54}$$

Situation 1. All boundary faces are assigned flux-type boundary conditions: If q_{AX}, q_{AY}, q_{AZ} represent the specified outward boundary fluxes through the three boundary faces and have been taken into account in computing the element residuals, Equation 54 becomes:

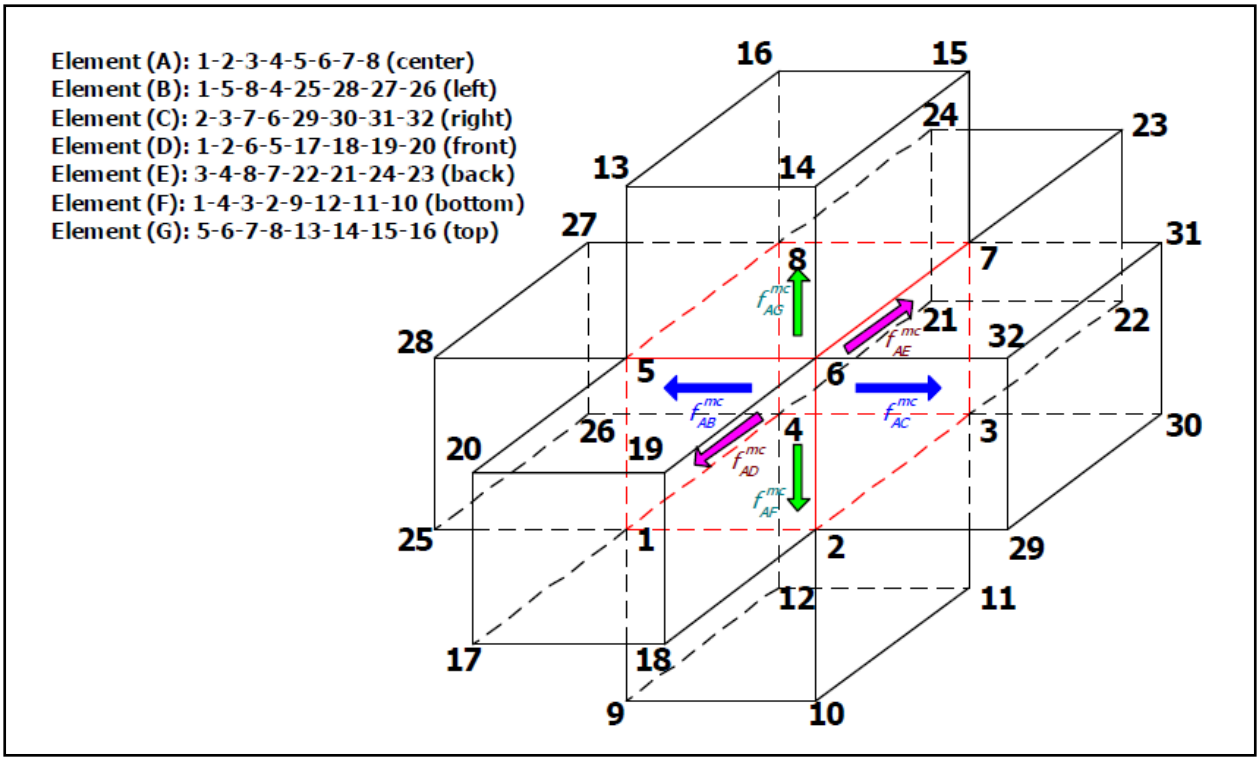

Figure 8. An internal hexadedral element (i.e., Element (A)) that has six internal faces.

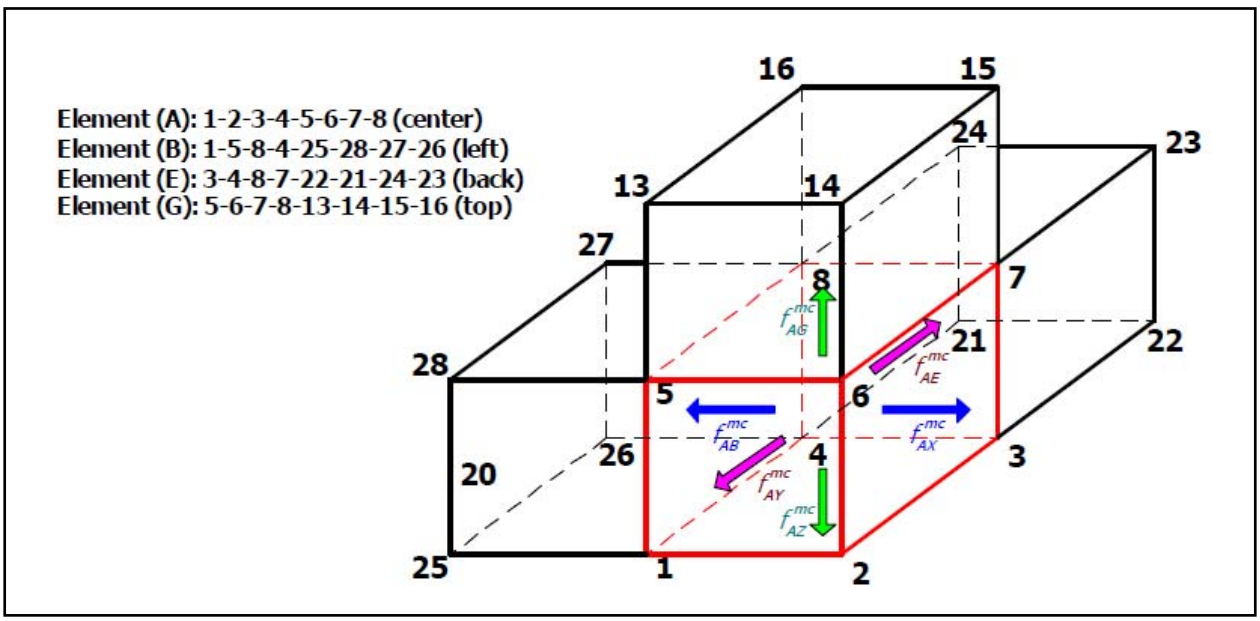

Figure 9. A boundary hexadedral element (i.e., Element (A)) that has three boundary and three internal faces.

$$\begin{aligned} R_A^{bf} &= R_A - q_{AX} - q_{AY} - q_{AZ} = f_{AB}^{mc} + f_{AE}^{mc} + f_{AG}^{mc} \\ &= \left[V_{AB} + \left(U_B - U_A\right)\right] \cdot A_{AB} + \left[V_{AE} + \left(U_E - U_A\right)\right] \cdot A_{AE} + \left[V_{AG} + \left(U_G - U_A\right)\right] \cdot A_{AG} \end{aligned} \tag{55}$$

The normal fluxes for Boundary Faces 2-3-7-6, 1-2-6-5, and 1-4-3-2 are computed with

$$V_{AX}^{mc} = q_{AX} \quad \text{(through Face 2-3-7-6)} \tag{56a}$$

$$V_{AY}^{mc} = q_{AY} \quad \text{(through Face 1-2-6-5)} \tag{56b}$$

$$V_{AZ}^{mc} = q_{AZ} \quad \text{(through Face 1-4-3-2)} \tag{56c}$$

Situation 2. Some, but not all, boundary faces are assigned flux-type boundary conditions: Suppose Face 2-3-7-6 is the only boundary face of Element (A) assigned an outward normal boundary flux, q_{AX}, the element residual equation for Element (A) is

$$\begin{aligned} R^{A,bf} &= R^A - q_{AX} = f_{AB}^{mc} + f_{AD}^{mc} + f_{AE}^{mc} + f_{AF}^{mc} + f_{AG}^{mc} \\ &= \left[V_{AB} + \left(U_B - U_A\right)\right] \cdot A_{AB} + \left[V_{AD} + \left(0 - U_A\right)\right] \cdot A_{AB} + \left[V_{AE} + \left(U_E - U_A\right)\right] \cdot A_{AE} \\ &\quad + \left[V_{AF} + \left(0 - U_A\right)\right] \cdot A_{AB} + \left[V_{AG} + \left(U_G - U_A\right)\right] \cdot A_{AG} \end{aligned} \tag{57}$$

The outward normal fluxes through Boundary Faces 2-3-7-6, 1-2-6-5, and 1-4-3-2 are computed by

$$V_{AX}^{mc} = q_{AX} \quad \text{(through Face 2-3-7-6)} \tag{58a}$$

$$V_{AY}^{mc} = V_{AY} + \left(0 - U_A\right) \quad \text{(through Face 1-2-6-5)} \tag{58b}$$

$$V_{AZ}^{mc} = V_{AZ} + \left(0 - U_A\right) \quad \text{(through Face 1-4-3-2)} \tag{58c}$$

Situation 3. None of the boundary faces is assigned a flux-type boundary condition: In this case, the outward normal fluxes can be computed by using

$$V_{AX}^{mc} = V_{AX} + \left(0 - U_A\right) \quad \text{(through Face 2-3-7-6)} \tag{59a}$$

$$V_{AY}^{mc} = V_{AY} + \left(0 - U_A\right) \quad \text{(through Face 1-2-6-5)} \tag{59b}$$

$$V_{AZ}^{mc} = V_{AZ} + \left(0 - U_A\right) \quad \text{(through Face 1-4-3-2)} \tag{59c}$$

Summary of conservative flux computation using the global approach. We've discussed how the element residual equation for an element, either internal or containing boundary faces, is composed in both two- and three-dimensional spaces. Suppose there are totally M elements in the FE computational mesh; we can construct M element residual equations, one for each element. These M element residual equations are assembled to form a

global matrix equation based on the face-element connectivity. This global matrix equation is then solved for the element-based *U*'s that are used to correct normal face flux for mass conservation.

SOFTWARE DEVELOPMENT: We developed a software toolkit named Consistent/inconsistent Conservative Flux Computation Toolkit (CCFlux) to assist application developers with computing locally conserved fluxes. Systems, e.g., the shallow water system, solved in the primitive form would preserve local mass conservation when fluxes are computed consistently. However, systems, such as subsurface flow, solved in a derived form without holding an explicit conservation law would not produce local-mass conserved fluxes because the flux is computed inconsistently with the discrete equation being solved. The goal is to obtain locally conserved flux across edges in 2-D or faces in 3-D systems. In the CCFlux toolkit, functionalities include (1) construction of a unique edges/faces map, (2) local renumbering of neighbor elements around a node on the index space, and (3) an efficient edge/face manipulation for divergence-free operation. The toolkit will be incorporated into different application codes, e.g., ADH (C code) [5], pWASH123D (mixed C and Fortran code) [6], or even the old-fashioned Fortran-77 FEMATER code [7]. Parallel mesh manipulation for flux calculation is embedded in CCFlux.

Presumably the mesh has a unique global element ID (GEid) assigned to each element shown in Figure 10, though each processor has its own local set of element IDs (LEid). This requirement defines unique flux direction systematically without exception in parallel simulation. In CCFlux, the first and second elements adjacent to an edge or a face store lower GEid and higher GEid, respectively, but perhaps lower LEid and higher LEid, respectively. However, the flux direction across a boundary edge is always defined from inside to outside of the domain. Therefore, the second element adjacent to a boundary edge is "-1," implying nonexistent.

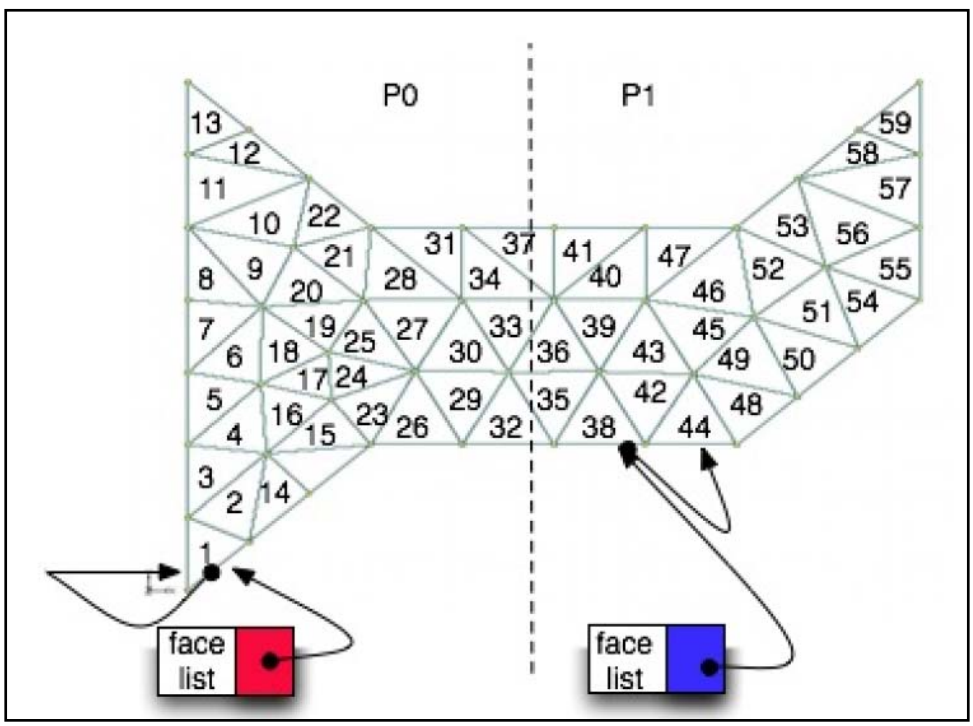

Figure 10. A partitioned 2-D domain showing global element ID and the edge list collecting unique edges on each processor.

In Figure 11, the red color arrows around vertex 32 show the flux direction associated with each adjacent sub-edge. The total flux of Edge 31-32 is the sum of two computed fluxes of this sub-edge based on node 32 and node 31. Note that vertex 32 and vertex 31 can be owned by different processors. The link list sketched at the bottom of Figure 11 collects all the unique edges on each processor. The pointer "Next" points to the next unique edge. However, "Next2[0]" and "Next2[1]" point to the next edge adjacent to the respective vertex comprising this edge. The first entry of "Next2" points to the next edge adjacent to the vertex with lower local vertex ID on the edge, while the second entry associated with the other vertex on this edge.

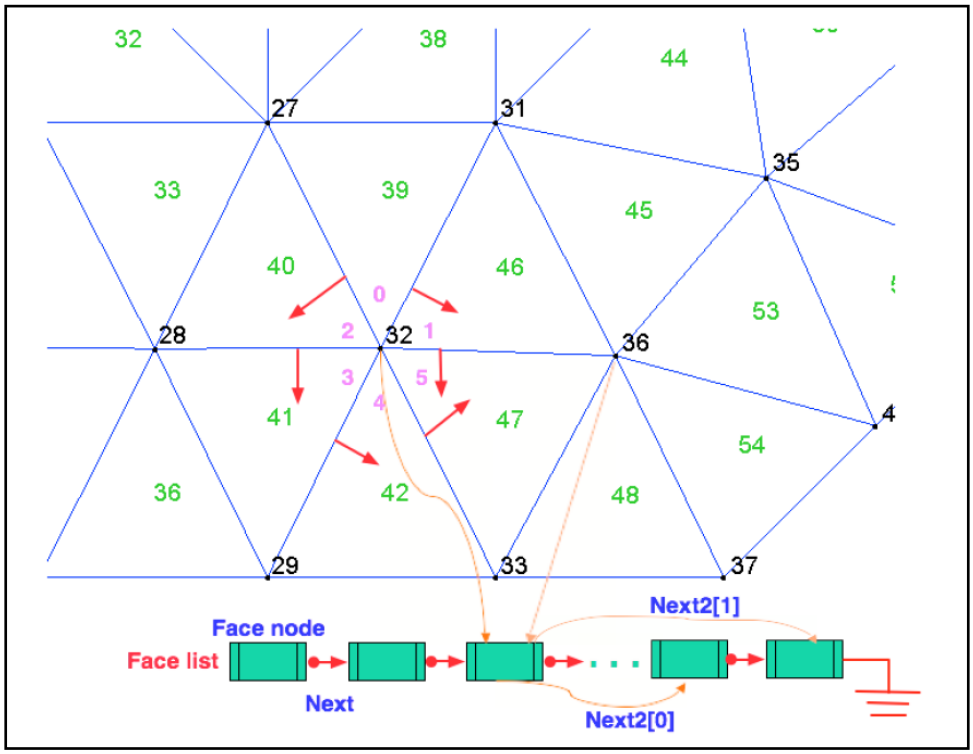

Figure 11. Three link lists collecting all unique edges, edges adjacent to vertex 32, and vertex 36.

This design of data structure avoids duplicate edge nodes, which can waste memory and CPU time spent in copying data among duplicate edge nodes. Each edge node has two data fields to store computed data defined by application codes. Each one is obtained based on each vertex on the edge. The CCFlux toolkit also renumbers the element ID adjacent to each vertex. For example, in Figure 11 those 6 elements adjacent to vertex are numbered to from 0 to 5. The toolkit can also provides a set of Boolean values indicating whether the adjacent flux direction of a node needs to be flipped if the circulation condition is imposed.

The toolkit equips a set of API functions listed in Table 1. The following three functions are designed for different application codes, and must be incorporated at the initialization stage. They are adhCCFinitMesh for ADH, pWASHCCFinitMesh for pWASH123D, and fwCCFinitMesh for FEMWATER. Three destroy functions, adhCCFdestroyMesh, pWASHCCFdestroyMesh, and fwCCFdestroyMesh are also required at the finalized stage. The data handler, CCFmesh, separates the flux calculation from others in the application to

make the algorithm implementation portable. The ultimate goal of the toolkit development—reusable and modular code—can thus be reached.

Table 1. API functions

API function	Return value*
face_node=CCF_GET_1ST_VTX_NEIGHBOR_FACE(mesh,i)	Return the 1st face (i.e., face node shown in Figure 11) adjacent to vertex i.
face_ptr=CCF_GET_FACE_POINTER(face_node)	Return the face pointer encapsulating the following information: face number (i.e. 0, 1, or 2 for a triangle) and the element pointer that the face_node resides.
ierr=CCFface_vertices(mesh,face_ptr,vtx_list)	Return the vertex list comprising the face that the face_ptr points to.
Sign_flipped=CCF_GET_FACE_SGN_DATA2VTX(mesh,i)	Indicating whether each flux direction across adjacent faces next to node i needs to be flipped if circulation condition is imposed.
data_ptr=CCFget_face_node_data(mesh,face_node)	Return the address pointing to user-defined data associated with the face_node.
CCFface_neighbor_elements(mesh,face_ptr,elmID,elmID+1)	Return the local element ID adjacent to the face that the face_ptr points to. The flux direction is defined from elmID[0] to elmID[1].
elm_face=CCFget_face_tuple(mesh,ptr,j)	Return the 2 adjacent compact element ID (0 to 5 show in Figure 11) of the jth face next to the vertex encapsulated by the pointer ptr.
face_node=CCFget_next_vtx_neighbor_face(mesh,face_ptr,face_node,elmID[0],i)	Return the next face_node of the current face_node retrieved from the face_list based on the info. of vertex i. Either the one linked by next2[0] or next2[1] is returned.
CCFsend(mesh)	Synchronize face data among processors

* "Face" is used to represent 2-D edge and 3-D face in this column

To demonstrate the algorithmic implementation, the local approach of Larson-Niklasson method is used to calculate the U values described from Equation 2 to Equation 10. Each U value is associated with an element adjacent to the vertex shown in Figure 2. For the 2-D shallow water flow in ADH, the estimated flux is the length integration of hV (a face is actually an edge in 2-D), shown in the first loop of the pseudo code in Figure 12. Followed is the loop visiting each vertex to construct a local linear system, in which the total number of rows equals the number of elements including the vertex. The LU direct solver is then used to solve this linear system with unknown U. Once U is obtained, the mass conserved flux, i.e., $(hV)^{mc}$, can be calculated based on the formulation from Equation 11a to Equation 11f. The final flux value across each face is actually the average of the local conserved $(hV)^{mc}$ associated with the vertex sitting on the edge. Before the computation of the final flux, the function CCFsend must be executed to synchronize the edge data $(hV)^{mc}$ for summing on the node star. To verify the result, we can simply compute the difference between the total flux across all the faces and the sum of boundary fluxes described in the governing equation. This verification function checks every element throughout the entire domain. The maximum value of the difference is then found and given to the user. Ideally it is zero but can practically be near the user's prescribed error tolerance for the application simulation.

```
Foreach face do
 data_ptr[0]=n•∫N[0](hV)dl
 data_ptr[1]=n•∫N[1](hV)dl
Endfor
Foreach owned vertex do
 pick elmID[0] or elmID[1] based on sign_flipped
 compute element residual, i.e., R1^A, …, R1^F
 construct A and b => AU = b
 set U1,A=0, LU direct solver is used to obtain sol.
 compute (hV)^mc for each face
Endfor
CCFsend(mesh) /* sync face data */
```

Figure 12. Pseudo code for the Larson-Niklasson implementation.

The implementation on a 3-D mesh should be similar except that some geometric issues need to be tackled. The CCFlux toolkit needs to be flawless for both 2-D and 3-D meshes. Currently, we are in the middle of debugging 3-D cases. A new version of CCFlux will be released after the 3-D bug-free implementation is completed.

EXPERIMENTAL RESULTS: We demonstrate the locally mass-conservative flux calculation on an example used in the article by Tate et al. [8]. The domain of interest is San Diego Bay in CA, which covers an area of approximately 560 km^2 and has a mean depth of 6.5 m. This area was discretized with 10,999 triangular elements and 6,311 nodes, as shown in Figure 13. The simulation took the tide shown in Figure 14 as the driving force and used a time-step size of 360 sec. The total simulation time was 24 hours, which is about a diurnal cycle. The tidal boundary condition was applied to the nodes included in the section of open boundary (Figure 13). The rest of the domain boundary was set to closed boundary, i.e., no normal flow.

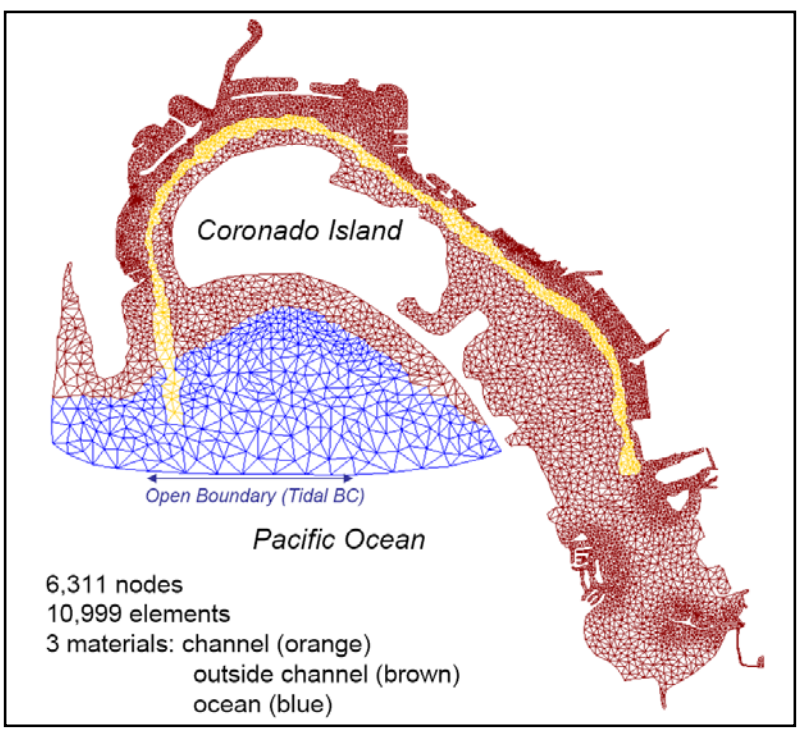

Figure 13. Domain discretization of San Diego Harbor.

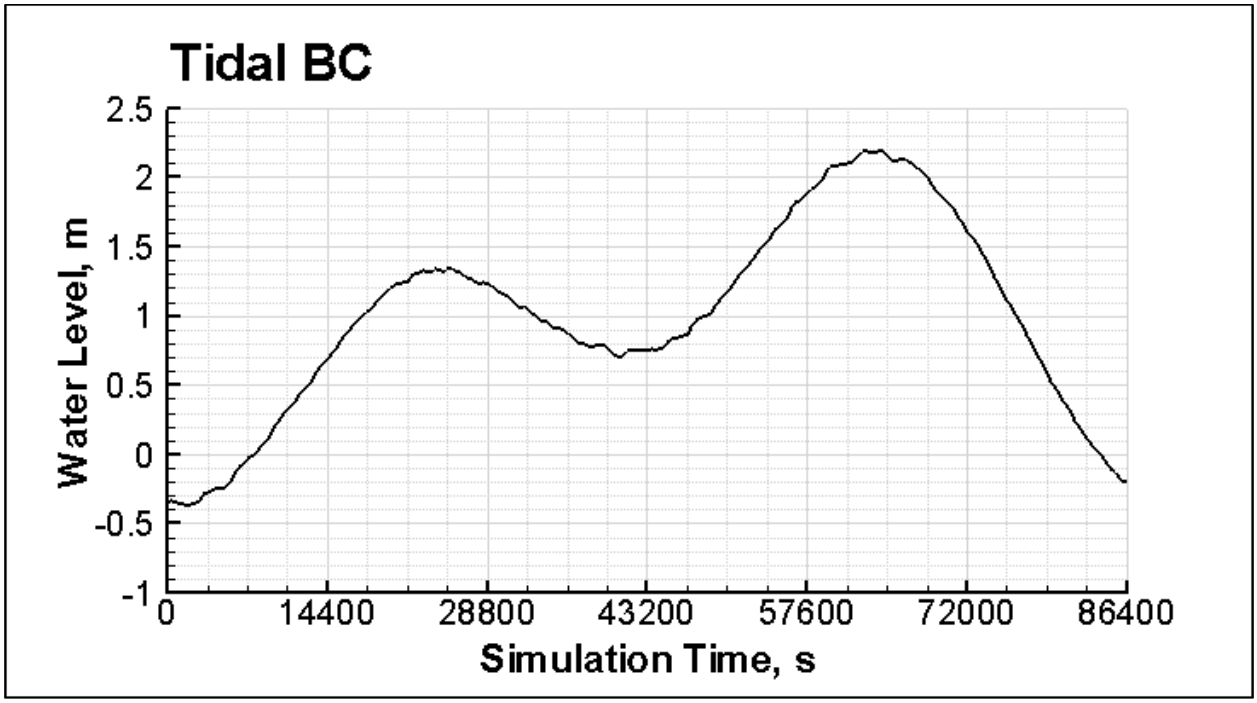

Figure 14. The tidal boundary condition.

We selected six nodes to examine the computed conservative fluxes. Among these six nodes, Nodes 21 and 22 are on the open boundary (Figure 15a), Nodes 4684 and 4685 are on the closed boundary (Figure 15b), and Nodes 405 and 2997 are interior nodes (Figure 15c). Figure 16 shows the flow direction, which is defined from the lower global element ID to the higher one. Figure 17 shows the water depth color contour and scaled velocity vector at simulation time = 3,600 s. Table 2 shows the estimated normal sub-edge flow (column 2), the conservative normal sub-edge flow (column 3) and the total nodal flow (column 4) at that time step. The flow rate shown in columns 2 and 3 are positive when the normal flow direction is the same as specified in Figure 16. The positive and negative signs associated with the total nodal flow in the "Nodal Flow" column represent the out-going and in-coming, respectively, flow at the nodes. As shown in Table 2, at simulation time = 3,600 the total nodal flow at Nodes 21 and 22 are negative, indicating incoming flow at those two open boundary nodes due to the rising tide (Figure 14). On the other hand, the total nodal flow at both the two interior nodes (i.e., Nodes 405 and 2997) and the two closed boundary nodes (i.e., Node 4684 and 4685) was zero, as expected in our derivation given in this technical note. Therefore, we have verified the implementation of the local approach for computing conservative normal fluxes on element edges in solving 2-D shallow water flow in ADH. The computational results show the maximum difference computed by the verification function mentioned in the section of "SOFTWARE DEVELOPMENT."

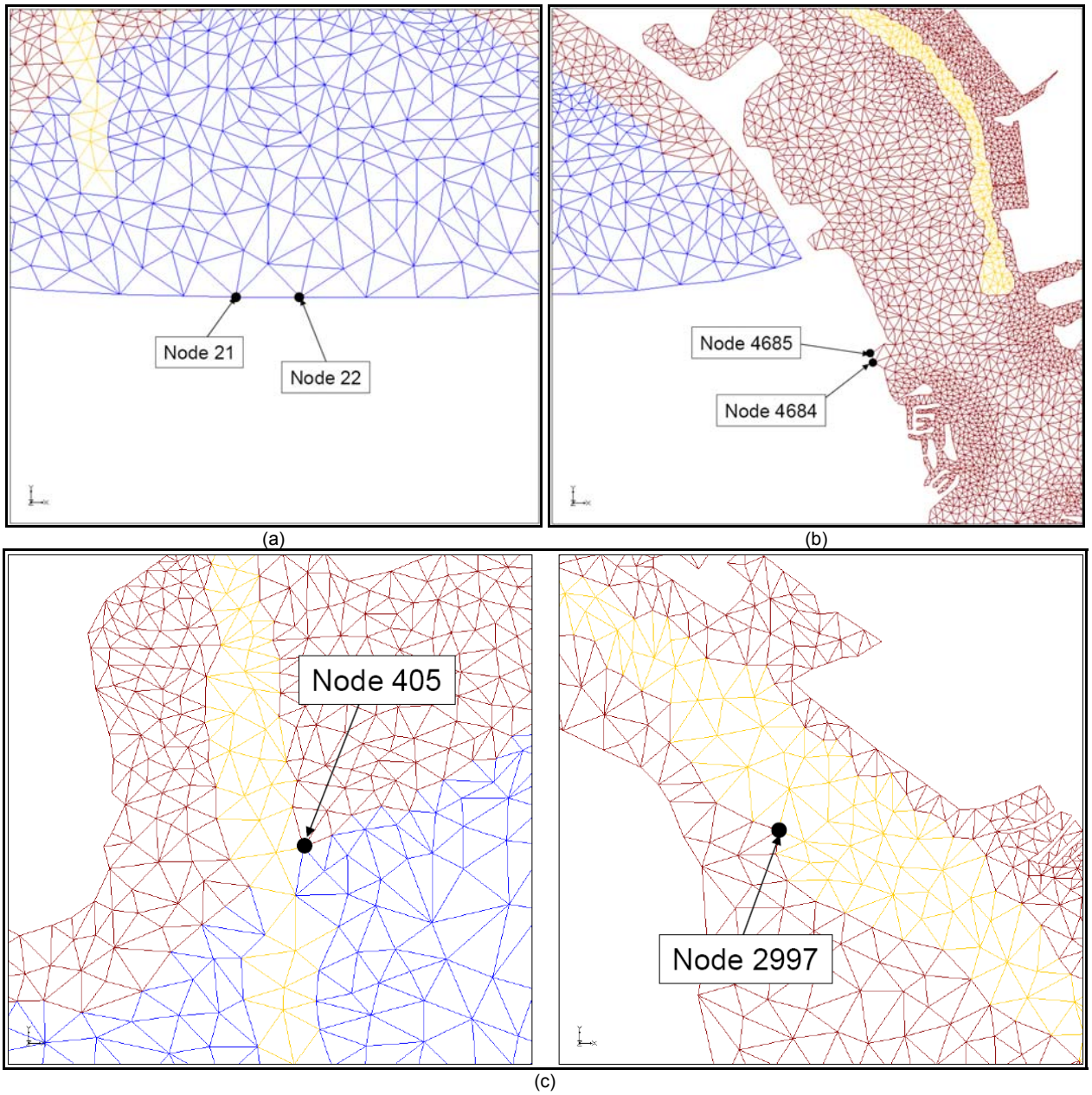

Figure 15. Locations of the six nodes selected to examine the computation of conservative flux: (a) Nodes 21 and 22 are on the open boundary, (b) Nodes 4684 and 4685 are on the closed boundary, and (c) Nodes 405 and 2997 are interior nodes.

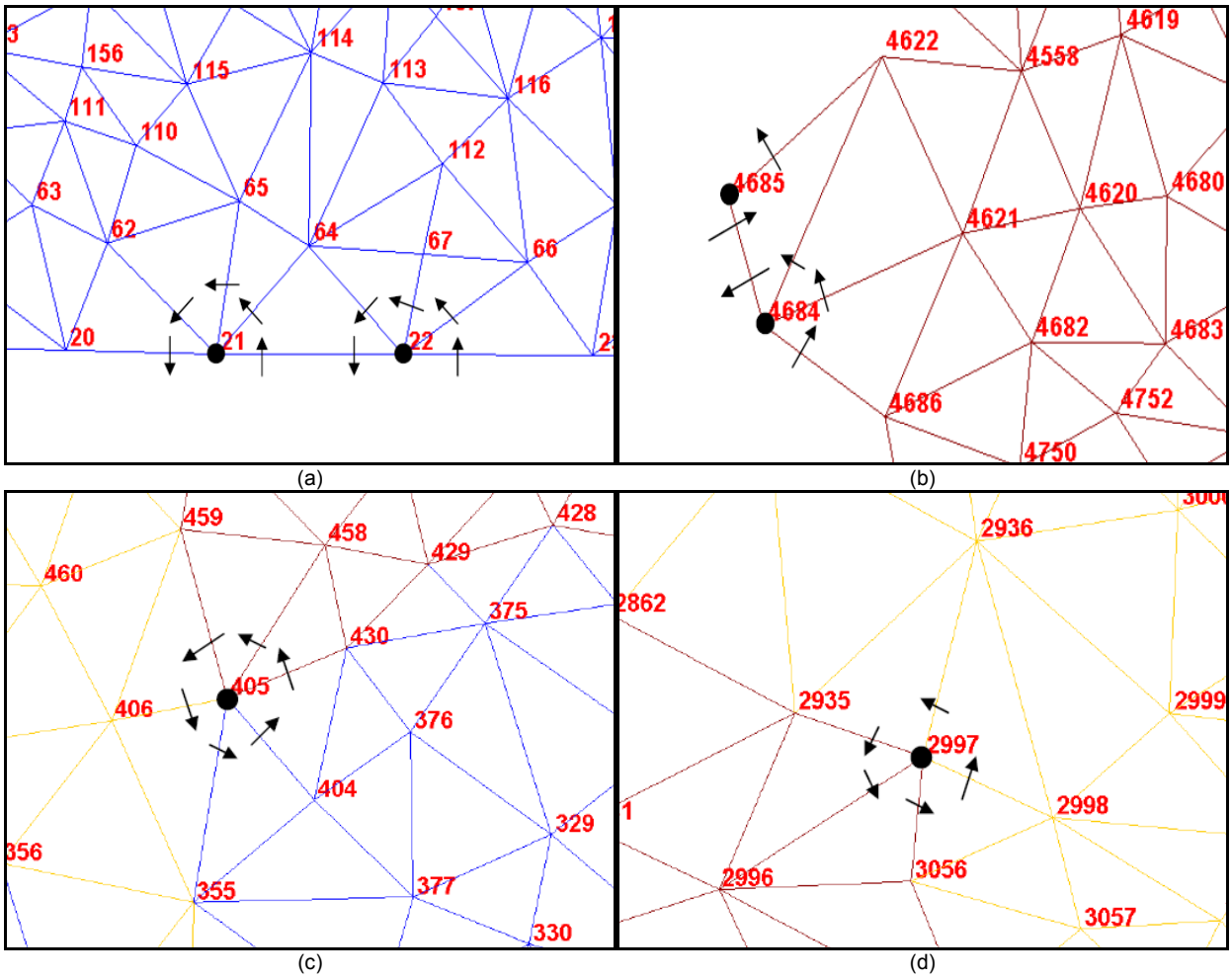

Figure 16. Zoom-in of the mesh discretization abound the six selected nodes: (a) Nodes 21 and 22, (b) Nodes 4684 and 4685, (c) Node 405, and (d) Node 2997; numbers in red are the global node IDs; black arrows are used to define positive flow direction across each sub-edge around the selected nodes.

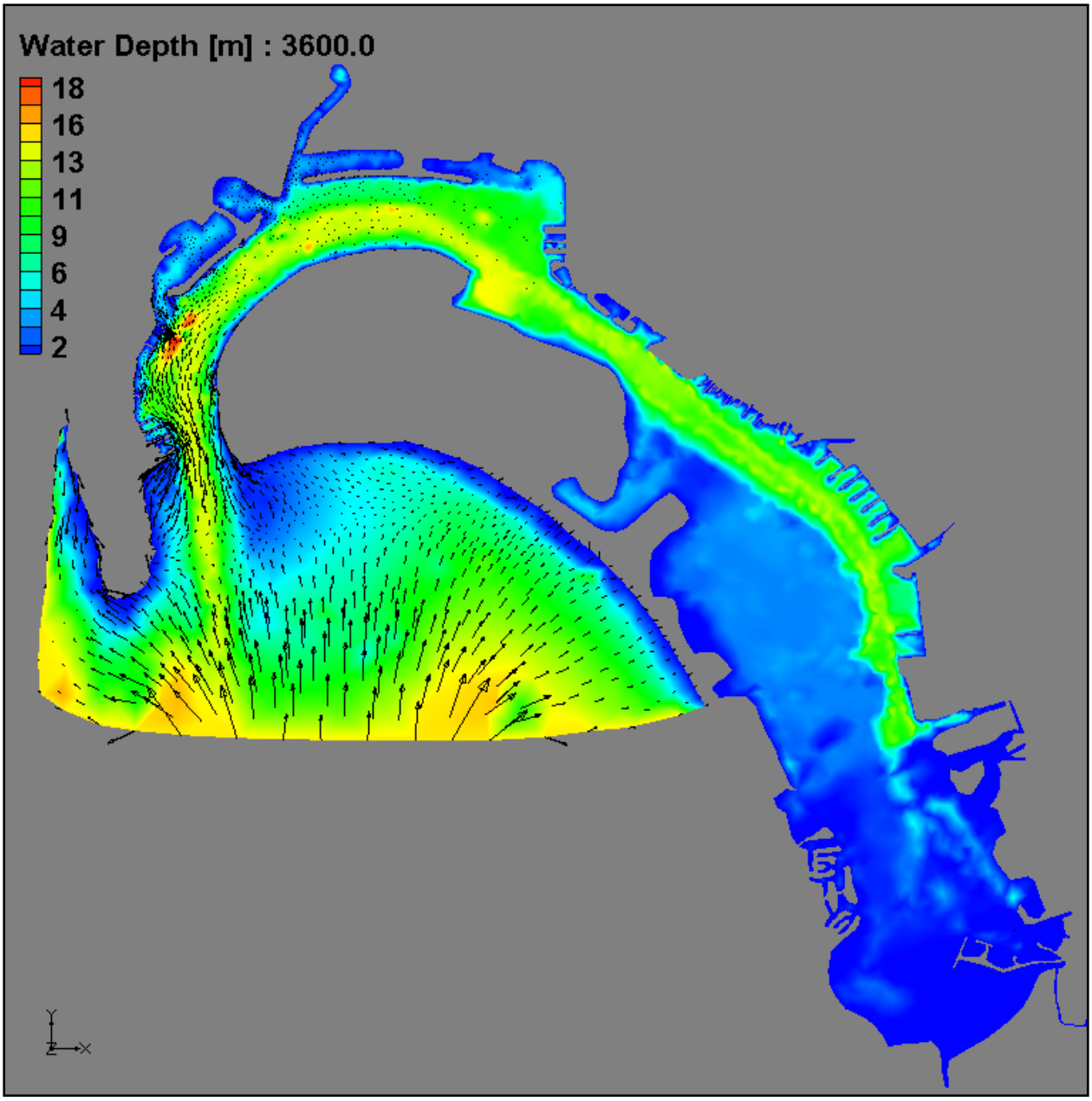

Figure 17. Computed water depth and velocity vector at simulation time = 3,600 s.

Table 2. Results of sub-edge and nodal flow associated with the six selected nodes at simulation time = 3,600 seconds

Sub-edge (two end nodes)	Estimated Flow* (ft^3/s)	Conservative Flow** (ft^3/s)	Nodal Flow*** (ft^3/s)
Around Node 21:			
21-22	2.7838E+04	2.7838E+04	-4.9348E+04
21-64	1.3247E+04	1.3162E+04	
21-65	3.3049E+03	3.1082E+03	
21-62	-1.5188E+04	-1.5625E+04	
21-20	-2.1510E+04	-2.1510E+04	
Around Node 22:			
22-23	3.4450E+04	3.4450E+04	-6.3437E+04
22-66	2.0206E+04	2.0516E+04	
22-67	2.4817E+03	2.8831E+03	
22-64	-1.5343E+04	-1.4908E+04	
22-21	-2.8987E+04	-2.8987E+04	
Around Node 405:			
405-404	4.8676E+02	4.8744E+02	0.0000E+00
405-430	6.1192E+02	6.3997E+02	
405-458	4.3372E+02	4.6648E+02	
405-459	-1.4027E+02	-1.6370E+02	
405-406	-7.7235E+02	-7.9921E+02	
405-355	-5.2722E+02	-5.3544E+02	
Around Node 2997:			
2997-2998	-3.3387E-02	-3.3632E-02	0.0000E+00
2997-2936	-1.2641E-01	-1.2681E-01	
2997-2935	3.6237E-02	3.7106E-02	
2997-2996	1.3369E-01	1.3483E-01	
2997-3056	4.8901E-02	4.7986E-02	
Around Node 4684:			
4684-4686	-4.3777E-06	0.0000E+00	0.0000E+00
4684-4621	-2.6924E-04	-6.3892E-03	
4684-4622	4.4110E-03	6.5927E-03	
4684-4685	-7.0438E-03	0.0000E+00	
Around Node 4685:			
4685-4684	7.6636E-03	0.0000E+00	0.0000E+00
4685-4622	3.7038E-03	0.0000E+00	

* Estimated flow = (computed water depth) x (computed velocity) x (sub-edge length)
** Conservative flow = (computed conservative flux) x (sub-edge length) using the local approach
*** Nodal flow = computational result by substituting the convergent solution into the matrix equation of mass conservation

SUMMARY: In this technical note, two algorithms to compute locally mass-conservative edge/face fluxes are presented. The local approach wins over the global approach and these were derived based on the Larson-Niklasson method. Obviously the global approach requires more memory consumption for the linear system, contrasted to the local approach, whose linear system size is defined by the total number of adjacent elements to a node. The global approach may require an efficient linear solver, while the local approach can simply use a direct solver.

Our software development aims to develop the CCFlux toolkit to support the solution of the local approach. A set of light-weight application interface functions facilitates easy incorporation to different applications. The aforementioned local approach was implemented in the 2-D shallow water flow module in the ADH model. An experimental case solving the San-Diego Bay area was presented for demonstration. The result verified the correct implementation of the locally-conservative flux computation and the CCFlux toolkit. More cases, including 3-D groundwater flow, 3-D CFD problems, and 3-D shallow water flow, will be built for demonstration in the near future.

ADDITIONAL INFORMATION: This work was funded by the SWWRP. At the time of publication, the SWWRP Program Manager was Dr. Steven L. Ashby: *Steven.L.Ashby@usace.army.mil*. This technical note should be cited as follows:

> Cheng, J-R. C., H-P. Cheng, M. W. Farthing, and C. E. Kees. 2010. *Computing locally-mass-conservative fluxes from multi-dimensional finite element flow simulations*. ERDC TN-SWWRP-10-4. Vicksburg, MS: U.S. Army Engineer Research and Development Center. *https://swwrp.usace.army.mil/*.

REFERENCES:

[1] Berger, R. C., and S. E. Howington. 2002. *Discrete Fluxes and Mass Balance in Finite Elements*, ASCE Journal of Hydraulic Engineering, 128 (1), 87-92.

[2] Kees, C. E., M. W. Farthing, and C. N. Dawson. 2008. *Locally conservative, stabilized finite element methods for variably saturated flow*, Comput. Methods Appl. Mech. Engrg. doi:10.1016/ j.cma. 2008.06.005.

[3] Larson, M., and A. Niklasson. 2004. *A conservative flux for the continuous Galerkin method based on discontinuous enrichment*, CALCOLO, 41, 65–76.

[4] Sun, S. Y., and M. F. Wheeler. 2006. *A posteriori error estimation and dynamic adaptivity for symmetric discontinuous Galerkin approximations of reactive transport problems*, Comput. Methods Appl. Mech. Engrg. 195 (7-8), 632–652.

[5] ADH. 2010. ADaptive Hydraulics Modeling. *https://adh.usace.army.mil/.*

[6] Yeh, G.-T., G. Huang, H.-P. Cheng, F. Zhang, H.-C. Lin, E. Edris, and D. Richards. 2006. *A First-Principle, Physics-Based Watershed Model: WASH123D*. Chapter 9, Watershed Models, Edited by V. P. Singh and D. K. Frevert. CRC Press, Taylor & Francis Group.

[7] Lin, H.-C., D. R. Richards, G.-T. Yeh, J.-R. C. Cheng, H.-P. Cheng, and N. L. Jones. 1997. *FEMWATER: A Three-Dimensional Finite Element Computer Model for Simulating Density-Dependent Flow and Transport in Variably Saturated Media*. Report CHL-97-12, U.S. Army Engineer Research & Development Center.

[8] Tate, J. N., R. C. Berger, and R. L. Stockstill. 2006. *Refinement Indicator for Mesh Adaption in Shallow-Water Modeling*. ASCE J. Hyd. Eng., Vol. 132, No. 8, 854-857, Aug 2006.

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