

Two-Sided Cut-Cell Boundary Method for Simulating Linear Terrain Features and 1D Stream Flows on a 2D Rectangular Mesh

Mustafa S. Altinakar¹, Marcus Z. McGrath¹, Yavuz Ozeren¹, Edie Miglio²

¹ Natl. Center for Computational Hydroscience and Engrg., The Univ. of Mississippi, Carrier Hall Room 102, P.O. Box 1848, University, MS 38677-1848, USA; PH: 1-662-915-3783; FAX: 1-662-915-7796; email: altinakar@ncche.olemiss.edu

² MOX Dept. of Math., Politecnico di Milano, Milano, Italy (Vis. Res. Scientist, NCCHE, The Univ. of Mississippi), Piazza Leonardo da Vinci, 32, 20133 Milano, Italy; PH: 39-02-23994600; FAX: 39-02-23994606; email: edie.miglio@polimi.it

ABSTRACT

A two-dimensional (2D) finite volume model that uses a first order upwinding scheme to simulate dam break floods over a complex topography, represented by a digital elevation model (DEM), was enhanced by implementing a two-sided cut-cell boundary method to represent linear terrain features, such as road and railroad embankments, which may significantly influence flood propagation and delineation. A special version of the cut-cell boundary is implemented to provide coupled 1D-2D simulation capability.

INTRODUCTION

The use of digital elevation model (DEM) as a regular rectangular computational mesh offers several advantages in two-dimensional (2D) flood simulations. The need for mesh generation is eliminated and the simulation results can be readily imported into GIS software to carry out risk and vulnerability analyses. Depending on the mesh size, however, the DEM may not resolve adequately certain linear terrain features, such as road and railroad, embankments, levees, etc., which may significantly affect the propagation of the flood and the extent of the inundated area.

The present paper describes the enhancement of an existing 2D shock capturing, conservative finite volume shallow-water model, which uses a cell-centered conservative upwinding scheme to simulate dam breaching floods over complex topography, by implementing a two sided cut-cell boundary to simulate the presence of important linear terrain features in the Cartesian mesh even if their width is too small to be represented in the DEM. Two-sided cut-cell boundaries can have flow on both sides. The overtopping may occur based on the water depths on both sides and the height of the cut-cell boundary.

A special version of the cut-cell boundary is also developed to represent a one-dimensional stream on a two-dimensional mesh. This capability allows coupled 1D-2D simulations of levee overtopping and/or breaching.

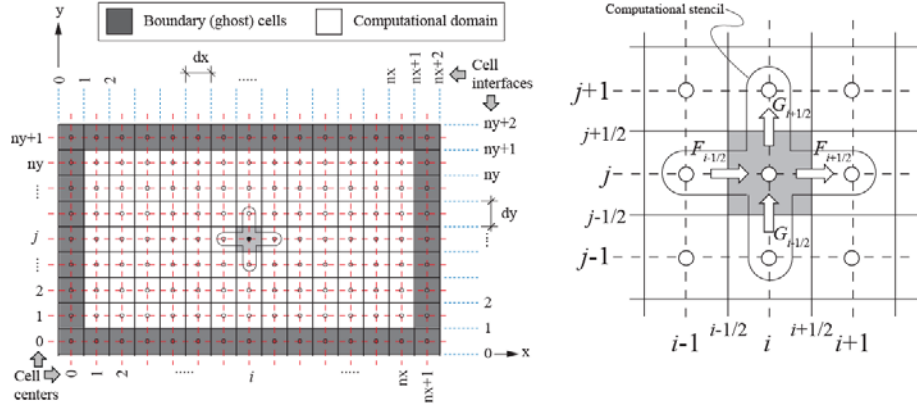


Figure 1. Cartesian computational mesh used in CCHE2D-FLOOD.

TWO-DIMENSIONAL MODEL: CCHE2D-FLOOD

Two-dimensional shallow water equations of flow in the horizontal plane can be written in vector form as follows:

$$\partial \mathbf{U} / \partial t + \partial \mathbf{F} / \partial x + \partial \mathbf{G} / \partial y = \mathbf{S} \quad (1)$$

The vectors of conserved variables, \mathbf{U} , fluxes in the x and y directions, $\mathbf{F}(\mathbf{U})$ and $\mathbf{G}(\mathbf{U})$, respectively, and sources, $\mathbf{S}(\mathbf{U})$, are defined as follows:

$$\mathbf{U} = \begin{bmatrix} h \\ Q_x \\ Q_y \end{bmatrix} \quad \mathbf{F} = \begin{bmatrix} Q_x \\ Q_x^2 / h \\ Q_x Q_y / h \end{bmatrix} \quad \mathbf{G} = \begin{bmatrix} Q_y \\ Q_y Q_x / h \\ Q_y^2 / h \end{bmatrix} \quad \mathbf{S} = \begin{bmatrix} 0 \\ -gh(\partial Z / \partial x) - g(u\sqrt{u^2 + v^2} / C^2) \\ -gh(\partial Z / \partial y) - g(v\sqrt{u^2 + v^2} / C^2) \end{bmatrix} \quad (2)$$

where h is the flow depth, u and v are the horizontal velocity components, $Q_x = uh$ and $Q_y = vh$ are the corresponding unit-width discharges. The water-surface elevation from a reference datum is Z , and the Chezy friction coefficient is C .

Consider the Cartesian regular computational mesh depicted in Figure 1. By integrating eq. 1 over a computational cell $\Delta x \times \Delta y$, and taking the time step to be Δt , the following finite volume discretization is obtained:

$$U_{ij}^{n+1} = U_{ij}^n - (\Delta t / \Delta x_i) (F_{i+1/2,j} - F_{i-1/2,j}) - (\Delta t / \Delta y_j) (G_{i,j+1/2} - G_{i,j-1/2}) + \Delta t S_{ij} \quad (3)$$

The intercell fluxes are computed using first order upwinding (Ying et al, 2004):

$$\mathbf{F}_{i+1/2j} = \begin{bmatrix} Q_x \\ Q_x^2 / h \\ Q_x Q_y / h \end{bmatrix}_{i+k} \quad k = \begin{cases} 0 & Q_x \geq 0 \\ 1 & Q_x \leq 0 \end{cases} \quad \mathbf{G}_{ij+1/2} = \begin{bmatrix} Q_y \\ Q_y Q_x / h \\ Q_y^2 / h \end{bmatrix}_{j+m} \quad n = \begin{cases} 0 & Q_y \geq 0 \\ 1 & Q_y \leq 0 \end{cases} \quad (4)$$

Wetting and drying is handled by defining a small water depth over the entire computational domain. Special algorithms are used for handling abrupt changes in the bottom topography. The computational domain, which can be a DEM, is surrounded by a single layer of ghost cells used for implementing various types of boundary conditions, such as: prescribed discharge inlet boundary, prescribed water-surface elevation boundary, fully reflecting wall, non reflecting outlet boundary, etc.

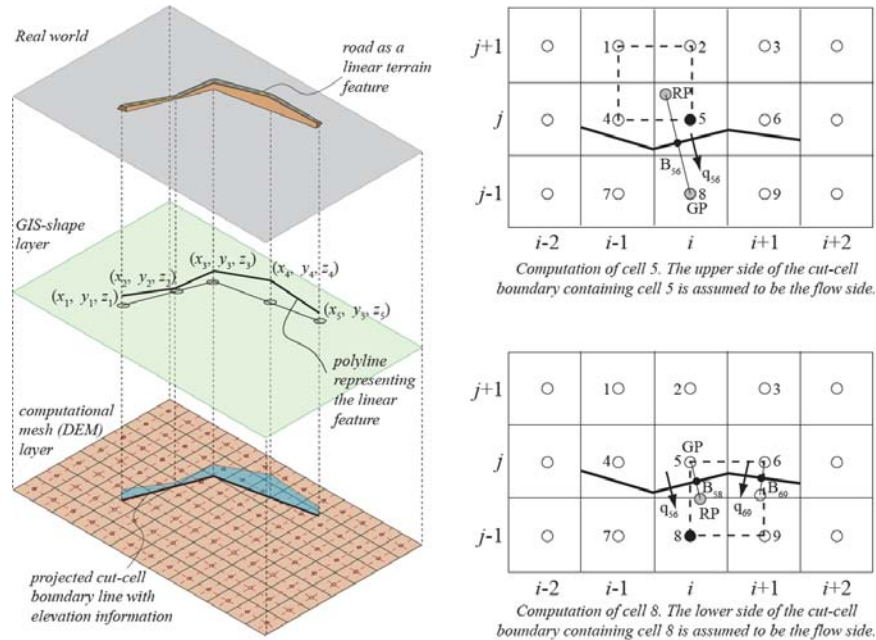


Figure 2. Projection of a road embankment onto the Cartesian mesh and the ghost-fluid method to compute cells affected by a cut-cell boundary.

CCHE2D-FLOOD is a robust and stable model and conserves mass rigorously. It has been verified using various analytical solutions and numerical tests, and validated by the laboratory field data available in the literature (Ying et al., 2003a and b; Ying and Wang, 2004; and Ying et al., 2004).

TWO-SIDED CUT-CELL BOUNDARY METHOD

Cut-cell boundary methods have been in use for defining irregular flow boundaries in a regular Cartesian mesh. Ingram et al. (2003), and Zhou et al. (2004) have used cut-cell boundaries in finite volume models for solving shallow water equations. In all these examples, the cut-cell boundaries are exclusively implemented to define an external or internal boundary with water on only one side.

In the present study, the cut-cell boundary method is used to represent linear terrain features, such as road and railroad embankments, which may be overtopped when the flow depth is sufficient. Considering also that the flow may be present on both sides, a two-sided cut-cell boundary approach was developed and implemented in CCHE2D-FLOOD. As it will be shown, presence of the water on both sides and the overtopping of the cut-line present special challenges that needed to be overcome.

Referring to Figure 2, the linear terrain feature is projected onto the DEM from a GIS shape layer by imposing that a cell can only be cut by a single straight line. Conserved variables in cells with one or more neighboring cell centers located on the other side of the cut-line cannot be updated using the computational stencil defined by eq. 3 (see Figure 1). The ghost fluid method by Ghias et al. (2007) was used to develop a special procedure to calculate these types of cells. This procedure does not require reducing the time step to meet the CFL condition for stability.

In the upper right hand corner of Figure 2, where some cells are numbered for ease of reference, the thick black line represents the cut-line projected onto the mesh. The neighboring cell 8 being on the other side of the cut-line, the computational stencil for cell 5 is incomplete. The cell 8 is, therefore, labeled as a *Ghost Point* (GP) with respect to cell 5. The cut-cell boundary method employs a two-step procedure. First, ghost fluid method is used to assign values to the conserved variables at GP to satisfy the following boundary conditions along the cut-line (boundary point B_{56}):

$$Q_n = q_{w_{56}} \quad ; \quad dQ_t / dn = 0 \quad ; \quad dZ / dn = 0 \quad (5)$$

where Q_n and Q_t are the normal and the tangential components of the discharge, $\mathbf{n} = [n_x, n_y]$ the vector normal to the cut-line and q the overtopping discharge. In the second step, the computational stencil is applied using the assigned values at GP.

To assign the values of the conserved variables at GP, the point GP is reflected to the other side of the cut-line to obtain the *Reflected Point* (RP), which is located in the rectangle formed by the cells 1, 2, 5, and 4. Any variable, ϕ , at point RP can be written using a bilinear interpolation based on the values in corners 1, 2, 5, and 4. The coefficients of the bilinear interpolation are obtained by solving the following system:

$$\begin{bmatrix} x_1 y_1 & x_1 & y_1 & 1 \\ x_2 y_2 & x_2 & y_2 & 1 \\ x_5 y_5 & x_5 & y_5 & 1 \\ x_4 y_4 & x_4 & y_4 & 1 \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \\ C_5 \\ C_4 \end{bmatrix} = \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_5 \\ \phi_4 \end{bmatrix} \quad (6)$$

Interpolating the values of $(Q_x)_{RP}$, $(Q_y)_{RP}$ at RP, the values of $(Q_x)_{GP}$, $(Q_y)_{GP}$ at GP are computed by a linear extrapolation. Note that:

$$(Q_n)_{RP} = (Q_x)_{RP} n_x + (Q_y)_{RP} n_y \quad \text{and} \quad (Q_t)_{RP} = -(Q_x)_{RP} n_y + (Q_y)_{RP} n_x \quad (7)$$

The first and second boundary conditions at point B_{56} respectively lead to:

$$(Q_n)_{GP} = 2q_{w_{56}} - (Q_n)_{RP} \quad (8)$$

$$(Q_t)_{GP} = (Q_t)_{RP} \quad (9)$$

The values of $(Q_x)_{RP}$, $(Q_y)_{RP}$ are obtained by solving eqs. 8 and 9 simultaneously. For depth at the RP, the imposition of the boundary condition $\partial Z / \partial n = 0$ implies that:

$$Z_{GP} = Z_{RP} \quad (10)$$

Knowing $(Q_x)_{GP}$, $(Q_y)_{GP}$ and Z_{GP} , the cell 5 is updated by using eq. 3.

When computing the cell, 8 as shown in the lower right hand side of Figure 2, the cell 5 becomes the GP. The RP of cell 5 is located in the rectangle formed by the cells 5, 6, 9, and 8. Two corners of the interpolation cell (5 and 6) are located on the other side of the cut-line with respect to cell 8 and cannot be used in the bilinear interpolation. The simple remedy is to replace the values of the missing corners 5 and 6 by the boundary conditions at B_{58} and B_{69} , respectively. Consider first the replacement of the corner 5 by the boundary condition at B_{58} . Using bilinear interpolation, the components of the discharge at a point (x, y) can be written as:

$$Q_x = C_{1x}xy + C_{2x}x + C_{5x}y + C_{4x} \quad (11)$$

$$Q_y = C_{1y}xy + C_{2y}x + C_{5y}y + C_{4y}$$

The first boundary condition (eq. 5) at point B_{58} can be written as written as:

$$\begin{aligned} & (C_{1x}x_{B_{58}}y_{B_{58}} + C_{2x}x_{B_{58}} + C_{5x}y_{B_{58}} + C_{4x})n_{x_{58}} + \\ & (C_{1y}x_{B_{58}}y_{B_{58}} + C_{2y}x_{B_{58}} + C_{5y}y_{B_{58}} + C_{4y})n_{y_{58}} = q_{w_{58}} \end{aligned} \quad (12)$$

Letting the tangent vector at B_{58} be defined as $\mathbf{t}_{B_{58}} = [-n_{y_{58}}, n_{x_{58}}]$, the second boundary condition (eq. 5) can be written as:

$$\left(\frac{\partial Q_t}{\partial n} \right)_{B_{58}} = \nabla(Q \cdot \mathbf{t}_{B_{58}}) \cdot \mathbf{n}_{B_{58}} = 0 \quad (13)$$

Using eq. 11 and simplifying, one finally obtains:

$$\begin{aligned} & [-(C_{1x}y_{B_{58}} + C_{2x})n_{y_{58}} + (C_{1y}y_{B_{58}} + C_{2y})n_{x_{58}}]n_{x_{58}} + \\ & [-(C_{1x}x_{B_{58}} + C_{3x})n_{y_{58}} + (C_{1y}x_{B_{58}} + C_{3y})n_{x_{58}}]n_{y_{58}} = 0 \end{aligned} \quad (14)$$

Eqs. 12 and 14 can be used to replace the missing corner 5 in the bilinear interpolation. A similar procedure is also used to obtain a pair of equations to replace the missing corner 6. Finally, the interpolation coefficients are obtained by solving the following system of equations:

$$\begin{bmatrix} (x_B y_B n_x)_{58} & (x_B y_B n_y)_{58} & (x_B n_x)_{58} & (x_B n_y)_{58} & (y_B n_x)_{58} & (y_B n_y)_{58} & n_{x_{58}} & n_{y_{58}} \\ (-y_B n_x n_y - x_B n_y^2)_{58} & (y_B n_x^2 + x_B n_x n_y)_{58} & -(n_x n_y)_{58} & (n_x n_x)_{58} & -(n_y n_y)_{58} & (n_x n_y)_{58} & 0 & 0 \\ (x_B y_B n_x)_{69} & (x_B y_B n_y)_{69} & (x_B n_x)_{69} & (x_B n_y)_{69} & (y_B n_x)_{69} & (y_B n_y)_{69} & n_{x_{69}} & n_{y_{69}} \\ (-y_B n_x n_y - x_B n_y^2)_{69} & (y_B n_x^2 + x_B n_x n_y)_{69} & -(n_x n_y)_{69} & (n_x n_x)_{69} & -(n_y n_y)_{69} & (n_x n_y)_{69} & 0 & 0 \\ x_9 y_9 & 0 & x_9 & 0 & y_9 & 0 & 1 & 0 \\ 0 & x_9 y_9 & 0 & x_9 & 0 & y_9 & 0 & 1 \\ x_8 y_8 & 0 & x_8 & 0 & y_8 & 0 & 1 & 0 \\ 0 & x_8 y_8 & 0 & x_8 & 0 & y_8 & 0 & 1 \end{bmatrix} \begin{bmatrix} C_{1x} \\ C_{1y} \\ C_{2x} \\ C_{2y} \\ C_{3x} \\ C_{3y} \\ C_{4x} \\ C_{4y} \end{bmatrix} = \begin{bmatrix} q_{w_{58}} \\ 0 \\ q_{w_{69}} \\ 0 \\ (Q_x)_9 \\ (Q_y)_9 \\ (Q_x)_8 \\ (Q_y)_8 \end{bmatrix} \quad (15)$$

The overtopping discharges appearing in eqs. 8 and 15 can be calculated using an appropriate weir equation. The user may choose between a trapezoidal road profile, which acts like a broad crested weir, and an ogee weir. In both cases, referring to Figure 3, the weir equation is given by:

$$Q = \sqrt{2g} C_e L H_u^{3/2} \quad (16)$$

where Q is the discharge passing over the weir of crest length L , C_e is the effective coefficient of discharge, and H_u is the upstream total head above the weir crest. The discharge coefficient for both types of weirs is given as a function of the parameter, P/H_u , where P is the height of the weir crest. Detailed information for ogee weirs and the computation of the discharge coefficient can be found in USBR (1987). Flow over road embankments were extensively studied by Kindsvater (1964). His results are summarized in Hamill (1999). It is assumed that the user does not have knowledge of the profile of the linear terrain feature and the available data is restricted to weir crest elevation and bed and water-surface elevations on both sides of the weir. Therefore, the effective discharge coefficient is calculated considering only the corrections due to downstream apron elevation and submergence. Other effects, such as the slope of the upstream face, roughness, etc., are neglected.

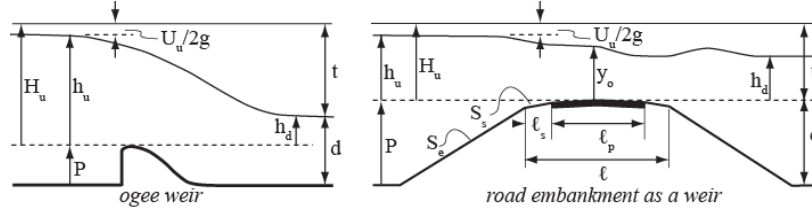


Figure 3. Definition sketches for weir flows.

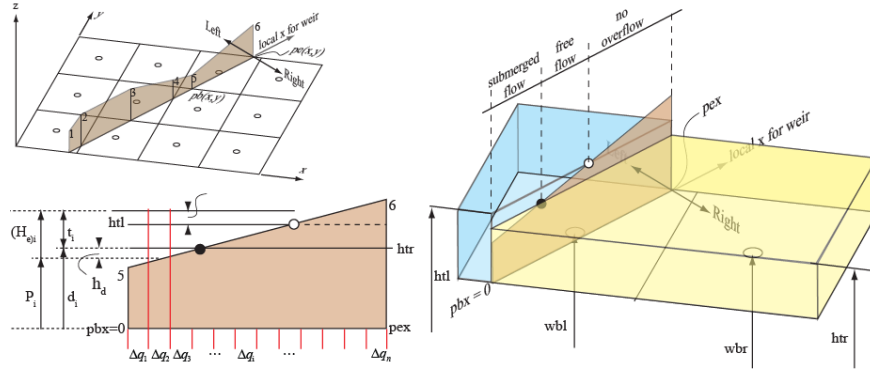


Figure 4. Implementation of weir flow calculation in CCHE2D-FLOOD.

Figure 4 shows the implementation of the weir flow calculation in CCHE2D-FLOOD. When a linear terrain feature is projected onto the Cartesian mesh, the intersections of the projected line with the gridlines are automatically computed, numbered, and stored in a database as the numbered vertices of a polyline, together with other useful geometric and relational information. In the most general case, three types of situations, i.e., submerged flow, free flow, and no flow, may coexist along the weir crest between two vertices. Therefore each weir portion between two vertices is automatically divided into a number of equal length segments. Considering local values of the parameters for each segment, the total unit discharge is computed from:

$$q_w = (1/L) \sum_{i=1}^n \Delta q_i = (1/L) \sum_{i=1}^n \sqrt{2g} (C_e)_i L_i (H_u)_i^{3/2} \quad \text{with} \quad L = \sum_{i=1}^n L_i \quad (17)$$

and stored in a database. At the next time steps, these values are directly used in eq. 8 or eq.15 for updating the values of the cells affected by cut lines. The results of test cases for verification and validation of the model with cut-cell boundary can be found in Miglio et al. (2008) and Altinakar et al. (2008).

COUPLED 1D-2D SIMULATIONS USING CUT-CELL BOUNDARIES

In recent years, there has been a growing interest in coupled 1D-2D simulations (Verwey; 2001; Lin et al., 2006). In the present study, the cut-cell boundary concept was extended to provide coupled 1D-2D simulation capability. The river is projected from a GIS-shape layer onto the Cartesian mesh (Figure 5) in the same way as a linear terrain feature, with the difference being that, now the river cross sections are also provided along with other data needed to construct a one-dimensional model of the river. A single line representing the thalweg line is projected onto the Cartesian mesh

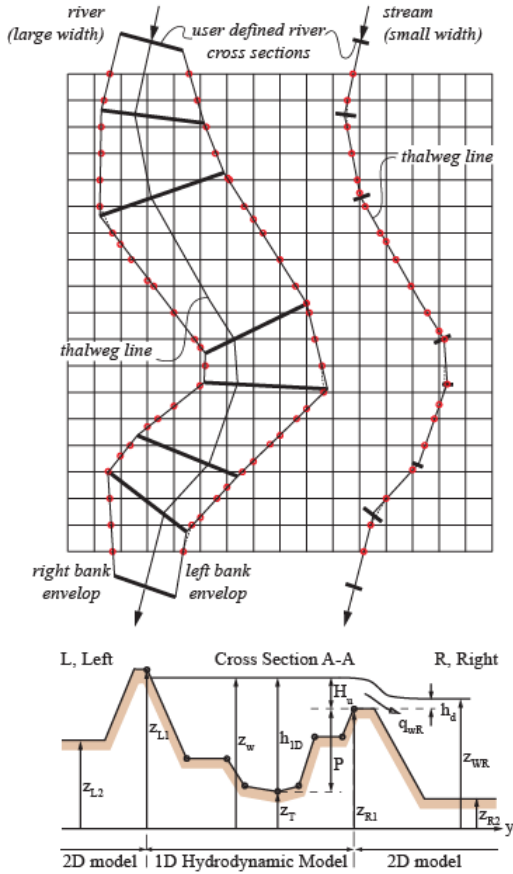
as a cut-cell boundary. The flow in the 2D model cannot cross this line. In this case the river cross sections are defined in the local coordinate system. If the river width is larger than the mesh size, the thalweg line and geo-referenced river cross sections must be provided. The pre-processor projects left and right envelop lines as cut-cell boundaries. The cut-lines act like barriers and the cells covered by the river are excluded from computation in the 2D model. 1D shallow-water equation is given by:

$$\partial \mathbf{U} / \partial t + \mathbf{F}(\mathbf{U}) / \partial x = \mathbf{S}(\mathbf{U}) \quad (18)$$

where vectors of conserved variables, intercell fluxes and source terms are given by:

$$\mathbf{U} = \begin{bmatrix} A \\ Q \end{bmatrix} \quad \mathbf{F}(\mathbf{U}) = \begin{bmatrix} Q \\ Q^2 / A \end{bmatrix} \quad \mathbf{S}(\mathbf{U}) = \begin{bmatrix} Q_t \\ -gh(\partial Z / \partial x) - g(n^2 |Q|Q| / R^{4/3} / A) \end{bmatrix} \quad (19)$$

The finite volume modeling of eq. 19 is described in Figure 5. In the current implementation, the 1D model is required to start outside of the 2D model. Various types of boundary conditions are available for the 1D model: inlet with hydrograph, inlet and outlet with specified head versus time curve, outlet with specified stage-discharge curve, etc. 1D and 2D models have their own time steps. At the beginning of each time step, the exchange discharges between 1D and 2D models are calculated using weir equation (eq. 17) based on the water surface elevations in the river (1D model) and the 2D model. Only the mass exchange is considered. The



Finite volume discretization of eq. 19 gives:

$$U_{ij}^{n+1} = U_{ij}^n - (\Delta t / \Delta x_i) (F_{i+1/2,j} - F_{i-1/2,j}) + \Delta t S_{ij}$$

Intercell fluxes are defined as:

$$\text{if } \begin{cases} Q_i^n > 0 \\ \text{and} \\ Q_{i+1}^n > 0 \end{cases} \Rightarrow F_{i+1/2} = \begin{bmatrix} Q_i \\ (Q_i)^2 / A_i \end{bmatrix}$$

$$\text{if } \begin{cases} Q_i^n < 0 \\ \text{and} \\ Q_{i-1}^n < 0 \end{cases} \Rightarrow F_{i+1/2} = \begin{bmatrix} Q_i \\ (Q_{i+1})^2 / A_{i+1} \end{bmatrix}$$

$$\text{otherwise } F_{i+1/2} = \begin{bmatrix} (Q_i + Q_{i+1}) \cdot V_f \cdot H_f / 2 \\ ((Q_i)^2 / A_i + (Q_{i+1})^2 / A_{i+1}) / 3 \end{bmatrix}$$

where

$$V_f = g \frac{\Delta t (z_{i+1} - z_i)}{x_{i+1} - x_i} \quad \text{and} \quad H_f = \frac{(H_i + H_{i+1})}{2}$$

The source term is given by:

$$S_i = \begin{bmatrix} Q_t \\ gA_i^{n+1} \left[\left(w_1 \frac{\Delta Z}{\Delta x} \right)_{up} + \left(w_2 \frac{\Delta Z}{\Delta x} \right)_{down} \right] - g \frac{(M_i^n)^2 |Q_i|}{(R_i)^{4/3} A_i} \end{bmatrix}$$

The net lateral discharge exchanged with 2D model is:

$$Q_t = q_{wR} + q_{wL}$$

Weighting factors are defined as:

$$w_1 = \min[1, C_r] \quad \text{and} \quad w_2 = \min[1, (1 - C_r)]$$

$$\text{with } C_r = \frac{\Delta t}{\Delta x} V_i^n$$

Figure 5. Projection of a river onto Cartesian mesh and 1D finite volume model.

momentum exchanges between 1D and 2D models are not taken into account. It is important to note that the user-specified river cross sections do not have to be aligned with any grid feature. During the pre-processing phase, when the river is projected onto the Cartesian mesh, a table of correspondence between river reaches and the 2D computational cells is created and written into an input file read by the program. This table is used when computing exchange discharges linking 1D and 2D computations. A test case for coupled 1D-2D simulation can be found in Altinakar et al (2008).

CONCLUSION

This paper describes a two-sided cut-cell boundary technique to represent linear terrain features, such as road and railroad embankments, in 2D shallow-water models even if they are not captured by the DEM due to their relatively small width. This two-sided cut-cell technique is based on ghost-cell method. The cut-cell boundary method was also extended to provide coupled 1D-2D simulation capability.

REFERENCES

- Altinakar, M., McGrath, M., Fijolek, E. and Miglio, E. (2008). “*Risk and Vulnerability Studies for Water Infrastructures Using a GIS-Based Decision Support System with 2D Numerical Flood Modeling*”, Proc. of the 8th Int. Conf. on Comput. Hydroscience and Engrg, ICHE-09, Sep. 8-12, Nagoya, Japan.
- Miglio, E. and Altinakar M. S. (2008). “*Representation of Linear Terrain Features in 2D Free Surface Models using Cut-Cell Boundary Method*”, Proc. of River Flow 2008, Int. Conf. on Fluvial Hydraulics, Sep. 3-5, Cesme, Izmir, Turkey.
- Ghias, R., Mittal, R. and Dong, H. (2007). “*A sharp interface immersed boundary method for compressible viscous flows*”, *J. Comput. Phys.*, 225 (1), 528-553.
- Hamill, L. (1999). *Bridge Hydraulics: Theory and Practice*, Taylor & Francis, 367pp.
- Ingram, D.M., Causon, D.M., Mingham, C.G. (2003). “Developments in Cartesian Cut Cell Methods.” *Mathematics and Computers in Simulation*, 61(3-6), 561-572.
- Lin, B., Wicks, J. M., Falconer, R. A. and Adams, K. (2006). “*Integrating 1D and 2D hydrodynamic models for flood simulation*”, Proceedings of the Institution of Civil Engineers Water Management, V. 159, Issue WM1, March, 19–25.
- USBR (1987). *Design of Small Dams*, A Water Resources Technical Publication - US Bureau of Reclamation, 3rd Ed., 904 pp. A PDF version can be downloaded at: http://www.usbr.gov/pmts/hydraulics_lab/pubs/manuals/SmallDams.pdf
- Verwey, A. (2001). “*Latest Developments in Floodplain Modelling -1D/2D Integration*”, keynote lecture in: *Proceedings of the 6th Conf. on Hydraulics in Civil Engineering*, Hobart, Australia, Nov. 28-30.
- Ying, X., Khan, A. A. and Wang, S.S.Y. (2004). “An Upwind Conservative Scheme for Saint-Venant Equations.” *Journal of Hydraulic Engineering*, ASCE, 130(10), 977-987.
- Zhou, J.G., Causon, D.M., Mingham, C.G., and Ingram, D.M. (2004). “Numerical Prediction of Dam-Break Flows in General Geometries with Complex Bed Topography.” *Journal of Hydraulic Engineering*, ASCE, 130(4), 332-340.