 Finite Volume Model for Two-Dimensional Shallow Water Flows on Unstructured Grids

Tae Hoon Yoon, F ASCE, 1 and Seok-Koo Kang 2

Abstract: A numerical model based upon a second-order upwind finite volume method on unstructured triangular grids is developed for solving shallow water equations. The HLL approximate Riemann solver is used for the computation of inviscid flux functions, which makes it possible to handle discontinuous solutions. A multidimensional slope-limiting technique is employed to achieve second-order spatial accuracy and to prevent spurious oscillations. To alleviate the problems associated with numerical instabilities due to small water depths near a wet/dry boundary, the friction source terms are treated in a fully implicit way. A third-order total variation diminishing Runge–Kutta method is used for the time integration of semidiscrete equations. The developed numerical model has been applied to several test cases as well as to real flows. Numerical tests prove the robustness and accuracy of the model.

DOI: 10.1061/(ASCE)0733-9429(2004)130:7(678)

CE Database subject headings: Floods; Hydraulic models; Numerical models; Shallow waters; Unsteady flow.

Introduction

Flows in rivers, floodplains, and coastal zones are very complex due to uneven bottom topography and irregular boundaries of the flow domain. These cannot be easily solved by the unidimensional models and even the bidimensional models cannot produce accurate results if they are not able to handle complicated geometries or are not robust enough to treat abrupt flow changes such as shocks and discontinuities or dry bed conditions. Several numerical schemes have been developed to alleviate the drawbacks involved in the bidimensional models but it seems that there exists no an all-around model so far.

Two-dimensional shallow water equations have been widely used to simulate flows in shallow lakes, wide rivers, estuaries, and coastal zones. A number of numerical methods have been developed to solve these equations, such as the finite difference method (Garcia and Kahawita 1986; Fennema and Chaudhry 1990; Molls and Chaudhry 1995), the finite element method (Akanbi and Katopodes 1988), and the finite volume method (Alcrudo and Garcia-Navarro 1993; Zhao et al. 1994; Anastasiou and Chan 1997; Sleigh et al. 1998). The finite difference methods have been used with structured grids that permit the flow field to be solved efficiently. Structured grids, which make the flow solver more efficient, may confront difficulties in modeling complex flow geometries. In these cases, unstructured grids can alleviate the problems associated with structured grids. A triangular mesh is generally the simplest and most convenient way for covering a two-dimensional domain. An advantage of using triangular grids is their ability to generate grids on arbitrary geometries and to increase the number of cells in high-gradient regions or in regions of particular interest in the flow field. It is a quite attractive technique for modeling rivers or coastal zones because the complex geometry is intractable when using structured grids.

Recently, several successful schemes have been presented to solve the shallow water equations on unstructured grids by using finite volume formulations. Anastasiou and Chan (1997) and Sleigh et al. (1998) reported a solution of the two-dimensional shallow water equations using second-order finite volume methods on triangular meshes. Zhao et al. (1994) developed a finite volume model with first-order spatial accuracy on unstructured meshes. Currently, there has not been much work done toward applying unstructured finite volume methods to real flow problems and verifying their capability to predict the real flow field. Zhao et al. (1994) applied their model to the Kissimmee River basin in the United States reporting satisfactory results. Sleigh et al. (1998) applied a second-order finite volume method to the Axe estuary in the United Kingdom.

In an attempt to circumvent many difficulties present in the existing numerical models and to verify the applicability of the finite volume method to real flow problems, a two-dimensional model is proposed and tested in this paper. The model is based on the upwind finite volume method on unstructured triangular grids and employs a cell-centered finite volume formulation to solve conservative two-dimensional shallow water equations. In order to achieve high-order spatial accuracy and to prevent nonphysical oscillations, the multidimensional reconstruction technique and the continuously differentiable multidimensional limiter proposed by Jawahar and Kamath (2000) are employed in this study. Jawahar and Kamath (2000) applied the reconstruction technique and the limiter to solve the two-dimensional linear advection equation and compressible Euler and Navier–Stokes equations. The reconstruction technique is based on a wide computational stencil and

1Professor Emeritus, Han River Eco-Hydro Inst., 367-6 Chunma-Oksoo Bldg. 3F, Oksoo-dong, Sungdong-gu, Seoul, Korea 133-100; formerly, Dept. of Civil Engineering, Hanyang Univ., Seoul, Korea. E-mail: taehyoon@hotmail.com
2Researcher, Hyundai Institute of Construction Technology, Hyundai Engineering & Construction, Co. Ltd., 102-4, Mabuk-ri, Goosung-eup, Yongin, Kyong-gi-do, Korea 449-716. E-mail: kangsk78@ihanyang.ac.kr

Note. Discussion open until December 1, 2004. Separate discussions must be submitted for individual papers. To extend the closing date by one month, a written request must be filed with the ASCE Managing Editor. The manuscript for this paper was submitted for review and possible publication on September 27, 2003; approved on September 25, 2003. This paper is part of the Journal of Hydraulic Engineering, Vol. 130, No. 7, July 1, 2004. ©ASCE, ISSN 0733-9429/2004/7-678–688/$18.00.
does not strongly depend on vertex values to preserve stability for highly distorted grids. The limiter is continuously differentiable and produces a smooth transition between discontinuous jumps with first-order accuracy and sharp but continuous gradients with second-order accuracy.

To accomplish high-order temporal accuracy, the time discretization is made by the third-order total variation diminishing (TVD) Runge–Kutta method (Shu and Osher 1988). The HLL approximate Riemann solver enabling one to handle discontinuous solutions is used to evaluate inviscid fluxes at the cell faces. The friction terms are treated fully implicitly by an operator splitting technique to prevent numerical instabilities caused by a small water depth near the dry zones.

**Governing Equations**

The two-dimensional depth-integrated shallow water equations are obtained by integrating the Navier–Stokes equations over the flow depth with the following assumptions: uniform velocity distribution in the vertical direction, incompressible fluid, hydrostatic pressure distribution, and small bottom slope. The continuity and momentum equations are

\[
\begin{align*}
\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} &= \mathbf{S} \\
\end{align*}
\]

in which

\[
\mathbf{U} = \begin{pmatrix} h \\ u h \\ v h \end{pmatrix}, \quad \mathbf{F} = \begin{pmatrix} u h \\ \frac{u^2 h + g h^2/2}{u} \\ \frac{v h}{u} \\ \frac{u^2 h + g h^2/2}{v} \end{pmatrix}, \quad \mathbf{G} = \begin{pmatrix} 0 \\ g h S_{\alpha x} \\ g h S_{\alpha y} \\ -g h S_{fx} \\ -g h S_{fy} \end{pmatrix}
\]

and

\[
\mathbf{S} = \mathbf{S}_0 + \mathbf{S} = \begin{pmatrix} 0 \\ g h S_{\alpha x} \\ g h S_{\alpha y} \\ -g h S_{fx} \\ -g h S_{fy} \end{pmatrix}
\]

where \( u \) and \( v \) = velocity components in the \( x \) and \( y \) directions, respectively; \( h \) = water depth; \( g \) = acceleration due to gravity; \( S_{\alpha x}, S_{\alpha y} \) = bed slopes in the \( x \) and \( y \) directions, and \( S_{fx}, S_{fy} \) = friction slopes in the \( x \) and \( y \) directions, respectively. In this study, the friction slopes are estimated by using the Manning formula

\[
S_{fx} = \frac{n^2 u \sqrt{u^2 + v^2}}{h^{3/2}}, \quad S_{fy} = \frac{n^2 v \sqrt{u^2 + v^2}}{h^{3/2}}
\]

where \( n \) = Manning’s roughness coefficient. In general, the influence of bottom roughness prevails over the turbulent shear stress between cells. Therefore the effective stress terms were neglected in the computation.

**Numerical Model**

A cell-centered finite volume method is formulated for Eq. (1) over a triangular control volume (Fig. 1), where the dependent variables of the system are stored at the center of the cell and represented as piecewise constants. Integrating Eq. (1) over the area of the \( i \)th control volume, one obtains

\[
\int_{A_i} \frac{\partial \mathbf{U}}{\partial t} dA + \int_{\Gamma_i} \mathbf{G} \cdot \mathbf{n} d\Gamma = \int_{\Delta A_i} \mathbf{S} d\Delta A
\]

where \( \Gamma_i \) = boundary of the \( i \)th control volume and \( \mathbf{n} \) is the unit outward vector normal to the boundary. Approximating the line integral by a midpoint quadrature rule, Eq. (4) can be written as

\[
\frac{d U_i}{d t} = -\frac{1}{A_i} \sum_{j=1}^{3} \mathbf{E}^* \cdot \mathbf{n}_j \Delta \Gamma_{ij} + \mathbf{S}_i
\]

where \( i \) and \( j \) denote the \( i \)th cell and the \( j \)th edge of the cell, respectively; \( U_i \) and \( S_i \) are the average quantities stored at the center of the \( i \)th cell; \( \mathbf{n}_j \) is the unit outward normal vector at the \( j \)th edge; \( \Delta \Gamma_{ij} \) is the length of the \( j \)th edge; and \( \mathbf{E}^* \) is the numerical flux through the edge which is computed by an exact or approximate Riemann solver.

The present model employs the HLL Riemann solver (Harten et al. 1983) to compute the normal flux at the face of a control volume as follows:

\[
\mathbf{E}^* \cdot \mathbf{n} = \begin{cases} 
(E_L)_{i,j} \cdot \mathbf{n} & \text{if } S_L \geq 0 \\
S_R (E_L)_{i,j} \cdot \mathbf{n} - S_L (E_R)_{i,j} \cdot \mathbf{n} + S_R S_L [ (U_R)_{i,j} - (U_L)_{i,j} ] & \text{if } S_L \leq 0 \leq S_R \\
(E_R)_{i,j} \cdot \mathbf{n} & \text{if } S_R \leq 0
\end{cases}
\]

where \((U_L)_{i,j}\) and \((U_R)_{i,j}\) = reconstructions of \( U \) on the left and right sides, respectively, and \( S_L \) and \( S_R \) = wave speed estimates.
There are several possible choices for $S_L$ and $S_R$. The approach proposed by Toro (1992) is used in the present model:

\[
S_L = \begin{cases} 
\min(q_L \cdot n - \sqrt{g h_L} u^* - \sqrt{g h_R} u^*) & \text{if both sides are wet} \\
q_L \cdot n - \sqrt{g h_L} & \text{if the right side is dry} \\
q_R \cdot n - 2 \sqrt{g h_R} & \text{if the left side is dry} 
\end{cases}
\]

\[
S_R = \begin{cases} 
\max(q_R \cdot n - \sqrt{g h_R} u^* + \sqrt{g h_L} u^*) & \text{if both sides are wet} \\
q_L \cdot n + 2 \sqrt{g h_L} & \text{if the right side is dry} \\
q_R \cdot n + \sqrt{g h_R} & \text{if the left side is dry} 
\end{cases}
\]

\[
u^* = \frac{1}{2} (q_L + q_R) \cdot n + \frac{1}{4} (q_L - q_R) \cdot n 
\]

\[
\sqrt{g h^*} = \frac{1}{2} (\sqrt{g h_L} + \sqrt{g h_R}) + \frac{1}{4} (q_L - q_R) \cdot n 
\]

where $\mathbf{q} = [u \ v]^T$.

**Linear Reconstruction**

Accuracy, which is the most important aspect for any flow solver, has a direct influence on the number of computational cells required to resolve a flow field as economically as possible. Representing the numerical approximation of the solution as a piecewise constant is equivalent to a first-order spatial accuracy, which is often inadequate to achieve a desired accuracy. For this reason, a higher-order implementation, which involves a gradient-reconstruction, is necessary.

There are several higher-order schemes applied to shallow water equations on unstructured triangular grids (Anastasiou and Chan 1997; Sleigh et al. 1998; Hubbard 1999; Wang and Liu 2000). A high-order reconstruction on unstructured grids is not an easy task since the positions of vertices and the numbers of elements surrounding a vertex are arbitrary. Extensions of one-dimensional reconstruction techniques, such as the MUSCL approach, to unstructured grids make the scheme strongly dependent on grid connectivity, and therefore poor results are obtained on highly distorted grids. In the present study, the reconstruction technique proposed by Jawahar and Kamath (2000), which possesses dependence on a wide computational stencil and does not strongly depend on vertex values to preserve stability for highly distorted triangles, is employed.

The initial data at each time step are reconstructed in the form

\[
U_i^{\text{new}} = U_i + r_i \cdot \nabla U_i 
\]

where $r_i$ is a position vector relative to the centroid of the cell and $\nabla U_i$ is the gradient.

For examples, the gradient of $U$ for the two triangles $\Delta 1a2$ and $\Delta 1i2$ in Fig. 1, which are referred to as $(\nabla U)_{1a2}$ and $(\nabla U)_{1i2}$, can be computed using the Green–Gauss theorem

\[
\nabla U = \frac{1}{A} \oint_{\Gamma} U \mathbf{n} d\Gamma 
\]

where $\Gamma$ is integration path connecting vertices of each triangle and $A$ is the area of the triangle.

In order to calculate gradients $(\nabla U)_{1a2}$ and $(\nabla U)_{1i2}$, a value of a conserved variable at a vertex should be known. For this purpose, a linearity preserving interpolation method based on the pseudo-Laplacian formula proposed by Holmes and Connell (1989) is used to calculate vertex values. Assuming that vertex $k$ is surrounded by $M$ cells, the conserved variable at vertex $k$ is calculated as follows:

\[
U_k = \sum_{i=1}^{M} \frac{\omega_i}{\sum_{i=1}^{M} \omega_i} U_i. 
\]

where

\[
\omega_i = 1 + \lambda_i (x_i - x_k) + \lambda_i (y_i - y_k) 
\]

\[
\lambda_i = \frac{I_{xy} R_y - I_{yx} R_x}{I_{xx} I_{yy} - I_{xy} I_{yx}}, \quad \lambda_i = \frac{I_{xy} R_y - I_{yx} R_x}{I_{xx} I_{yy} - I_{xy} I_{yx}} 
\]

\[
I_{xx} = \sum_{i=1}^{M} (x_i - x_k)^2, \quad I_{yy} = \sum_{i=1}^{M} (y_i - y_k)^2 
\]

\[
I_{xy} = \sum_{i=1}^{M} (x_i - x_k) (y_i - y_k) 
\]

\[
R_x = \sum_{i=1}^{M} (x_i - x_k), \quad R_y = \sum_{i=1}^{M} (y_i - y_k) 
\]

After $(\nabla U)_{1a2}$ and $(\nabla U)_{1i2}$ are obtained, the gradient at the face $j = 1$ is computed by using the area-weighted average of the two triangles $\Delta 1a2$ and $\Delta 1i2$

\[
(\nabla U)_{1j2} = \frac{A_{1a2} (\nabla U)_{1a2} + A_{1i2} (\nabla U)_{1i2}}{A_{1a2} + A_{1i2}} 
\]

$(\nabla U)_{j=2}$ and $(\nabla U)_{j=3}$ are computed in the same manner.

The unlimited gradient for the cell $i$ is computed using the area-weighted average gradients at the three faces

\[
(\nabla U)_i = \frac{A_{1a2} (\nabla U)_{1a2} + A_{1b3} (\nabla U)_{1b3} + A_{1i3} (\nabla U)_{1i3}}{A_{1a2} + A_{1b3} + A_{1i3}} 
\]

**Multidimensional Limiter**

Higher order schemes often produce nonphysical oscillations near discontinuities; therefore, it is essential to suppress nonphysical oscillations by limiting the slope of the reconstructed variables. For unstructured grids, the limiter should be inherently multidi-
mensional in construction and one-dimensional implementation is not suitable. In addition, the limiter should be continuously differentiable since the use of a nondifferentiable function such as max and min may adversely affect the convergence of the solution to steady state. For this reason, the multidimensional limiter of Jawahar and Kamath (2000), which is continuously differentiable, is used in this study.

The procedure of Jawahar and Kamath (2000) consists of calculating the limited gradient \( \nabla U_i^l \) as follows:

\[
\nabla U_i^l = a_1 \nabla U_{i_1} + a_2 \nabla U_{i_2} + a_3 \nabla U_{i_3}
\]

(20)

where \( a_1, a_2, \) and \( a_3 \) are weights given by the multidimensional limiter function and \( \nabla U_{i_1}, \nabla U_{i_2}, \) and \( \nabla U_{i_3} \) are unlimited gradients of the three surrounding cells which are combined to produce the limited gradient \( \nabla U_i^l \). The weights are given by:

\[
a_1 = \frac{(g_a g_c + \varepsilon^2)}{(g_a^2 + g_b^2 + g_c^2 + 3 \varepsilon^2)}
\]

\[
a_2 = \frac{(g_a g_b + \varepsilon^2)}{(g_a^2 + g_b^2 + g_c^2 + 3 \varepsilon^2)}
\]

\[
a_3 = \frac{(g_a g_b + \varepsilon^2)}{(g_a^2 + g_b^2 + g_c^2 + 3 \varepsilon^2)}
\]

(21)

where \( g_a, g_b, \) and \( g_c \) are functions of the gradients of the surrounding cells given by the square of the \( L_2 \) norm, i.e., \( g_a = \| \nabla U_{i_1} \|^2, g_b = \| \nabla U_{i_2} \|^2, \) and \( g_c = \| \nabla U_{i_3} \|^2; \) and \( \varepsilon \) is a small number which is introduced to prevent indeterminacy caused by the vanishing of the three gradients in regions of uniform flow.

After applying Eq. (19) for every grid cell, the unlimited gradients are computed and substituted into Eq. (20) to obtain the limited gradients. Finally, variables for each cell are reconstructed by Eq. (11).

**Treatment of Source Terms**

Source terms in Eqs. (1) and (2) consist of the slope and friction terms, and the treatment of these terms exerts a great influence on the accuracy of the numerical scheme. In triangular meshes, bottom slopes are readily computed since three vertices of a triangle lie on the same plane unlike the vertices of a rectangular mesh. The equation of a plane containing three vertices of a triangle is expressed as

\[
z = c_1 x + c_2 y + c_3
\]

(22)

where \( c_1, c_2, \) and \( c_3 \) are constants and \( z \) is the bottom elevation. Substitution of values of \( x, y, \) and \( z \) at three vertices into Eq. (22) yields the following simultaneous equations for \( c_1, c_2, \) and \( c_3 : \)

\[
x p c_1 + y p c_2 + c_3 = z_p \quad (p = 1, 2, 3)
\]

(23)

where \( p \) denotes a vertex of an element.

After \( c_1, c_2, \) and \( c_3 \) are obtained by solving Eq. (23), bottom slopes are computed as

\[
(S_{ox}, S_{oy}) = (-\frac{\partial z}{\partial x}, -\frac{\partial z}{\partial y}) = (-c_1, -c_2)
\]

(24)

Therefore, slope source terms are obtained as

\[
S_o = \begin{pmatrix} 0 \\ -gh c_1 \\ -gh c_2 \end{pmatrix}
\]

(25)

In treating the friction source terms, a simple explicit discretization may cause numerical instabilities when the water depth is very small, which commonly occurs near wet/dry boundaries. To circumvent numerical instabilities, the friction terms are treated in a fully implicit way with an operator splitting technique.

Eq. (5) can be split into two ordinary differential equations

\[
\frac{dU_i}{dt} = S_{oi}
\]

(26)

\[
\frac{dU_i}{dt} = -\frac{1}{A_{ij}} \sum_{j=1}^{3} E_j n_j \Delta T_{ij} + S_{ai}
\]

(27)

The right-hand side of Eq. (26) contains only friction source terms. Eqs. (26) and (27) are solved in implicit and explicit ways, respectively. Eq. (26) is solved by a fully implicit scheme utilizing a Taylor series expansion about the \( n \)th time level at every cell \( i \) in the domain:

\[
U_i^{n+1} - U_i^n = \Delta t S_{oi}^{n+1}
\]

(28)

\[
S_{oi}^{n+1} = S_{oi}^n + \frac{\partial S_{oi}}{\partial U} \Delta U_i + O(\Delta U^2)
\]

(29)

where \( n \) denotes the time level and \( \Delta U_i = U_i^{n+1} - U_i^n \). The second term of the right-hand side of Eq. (29) is the jacobian of \( S_i \). After some algebraic manipulations, one obtains

\[
(I - \Delta t \frac{\partial S_{oi}^n}{\partial U}) \Delta U_i = \Delta t S_{oi}^n
\]

(30)

where \( I \) is the identity matrix.

The increment of \( U \) due to the friction term is calculated by solving Eq. (30).

**Time Integration**

Using the solution of Eq. (26) as an initial condition, Eq. (27) is solved by the total variation diminishing (TVD) Runge–Kutta time discretization method, which preserves strong stability properties of the backward Euler method and high-order accuracy. This was proven very useful in solving hyperbolic partial differential equations (Shu and Osher 1988). Denoting the first term of the right-hand side of Eq. (27) as \( L(U) \), an optimal third-order TVD Runge–Kutta method is given by

\[
U^{(1)} = U^n + \Delta t L(U^n)
\]

\[
U^{(2)} = \frac{3}{4} U^n + \frac{1}{4} U^{(1)} + \frac{1}{4} \Delta t L(U^{(1)})
\]

(31)

\[
U^{n+1} = \frac{1}{3} U^n + \frac{2}{3} U^{(2)} + \frac{2}{3} \Delta t L(U^{(2)})
\]

Since the TVD Runge–Kutta method is an explicit scheme, the time step is restricted by a Courant–Friedrichs Lewy (CFL)-like condition. The following formula is used to determine the maximum time step at each time level:

\[
\Delta t \leq \min \left( \frac{R_i}{2 \max_i (\sqrt{u^2 + v^2 + 2} c_i)} \right)
\]

(32)

where \( R_i \) denotes the nearest distance from the centroid of the control volume to cell vertices and \( c \) is the wave celerity. In Eq. (32), the minimum is taken over all the cells in the computational domain and the maximum is taken over the three adjacent cells of \( i \).
Boundary Conditions

Boundary conditions are imposed at the face of the cell and the values of the conserved variables at the face are extrapolated from the cell-center. According to the theory of characteristics, the Riemann invariants of the one-dimensional shallow water equations are

\[ R^- = u + 2c, \quad R^+ = u - 2c \]  

which are conserved along \( dx/dt = u + c \) and \( dx/dt = u - c \), respectively, when the contribution of the source terms are neglected. \( R^- \) and \( R^+ \) denote the state to the right and left of a face, respectively. Since the right side of a boundary is outside the domain, the \( R^- \) condition is replaced by the boundary condition itself. For two-dimensional shallow water equations the \( R^- \) condition is given as

\[
(u,v)_L \cdot n + 2 \sqrt{gh_L} = (u,v)_* \cdot n + 2 \sqrt{gh_*} \tag{34}
\]

where the subscripts * and L denote the variables at the boundary and the left side, respectively. Eq. (34) is combined with the boundary condition to compute the normal flux at the boundary. The normal flux at the boundary is given as

\[
E \cdot n = \begin{pmatrix} h_u(u,v)_* \cdot n \\ h_u u_u(u,v)_* \cdot n + \frac{1}{2} \sqrt{gh_u^2} n_x \\ h_u v_u(u,v)_* \cdot n + \frac{1}{2} \sqrt{gh_u^2} n_y \end{pmatrix} \tag{35}
\]

where \( n_x \) and \( n_y \) = components of \( n \) in the x and y directions.

According to the theory of characteristics, two boundary conditions are needed when the flow regime is subcritical. For a subcritical flow, a boundary condition is imposed in the form of flow depth, unit discharge, or velocity.

In the case of a depth boundary condition, \( h_* \) is given and \( (u,v)_* \cdot n \) is computed directly from Eq. (34) as

\[
(u,v)_* \cdot n = (u,v)_L \cdot n + 2 \sqrt{gh_L} - 2 \sqrt{gh_*} \tag{36}
\]

In the case of a velocity boundary condition, \( (u,v)_* \cdot n \) is given and \( h_* \) is computed by modifying Eq. (34)

\[
h_* = \frac{[(u,v)_L \cdot n + 2 \sqrt{gh_L} - (u,v)_* \cdot n]^2}{4g} \tag{37}
\]

A unit discharge boundary condition is given as

\[
q = h_u(u,v)_* \cdot n \tag{38}
\]

where \( q \) = unit discharge, which is constant at a given time stage. \( h_u \) and \( (u,v)_* \) are computed by combining Eqs. (34) and (38). The substitution of \( q/h_u = (u,v)_* \cdot n \) into Eq. (34) yields a nonlinear equation for \( h_u \) that can be solved by an iterative method such as the Newton–Raphson method.

Assuming that the tangential velocity at a boundary is equal to that of left state, the tangential velocity component at a boundary is given as

\[
(u,v)_* \cdot t = (u,v)_L \cdot t \tag{39}
\]

After \( (u,v)_* \cdot n \) and \( (u,v)_* \cdot t \) are computed as stated above, \( u_* \) and \( v_* \) can be computed as

\[
\begin{pmatrix} u_* \\ v_* \end{pmatrix} = \begin{pmatrix} n_x & -n_y \\ n_y & n_x \end{pmatrix} \begin{pmatrix} (u,v)_* \cdot n \\ (u,v)_* \cdot t \end{pmatrix} \tag{40}
\]

Finally, the normal flux, Eq. (35) can be computed by using \( u_* \), \( v_* \), and \( h_* \).

Applications

Oblique Hydraulic Jump

When a supercritical flow passes a channel with decreasing width, the deflected channel wall generates an oblique standing wave accompanying an abruptly increased flow depth. This phenomenon is called an oblique hydraulic jump, for which an analytical solution is available (Chow 1959). This problem is used to verify the capability of the present model to handle high-speed discontinuous flows and the stability of the present model on distorted grids.

The computational domain of the problem is shown in Fig. 2. The initial conditions are set as \( h = 1 \text{ m}, u = 9 \text{ m/s}, \) and \( v = 0 \). A supercritical inflow condition \( (h = 1 \text{ m}, u = 9 \text{ m/s}, Fr = 2.87) \) and a free outfall condition are applied at the upstream and downstream boundaries, respectively.

Numerical tests are performed on two different grids: well-connected and highly distorted grids. The well-connected grid is made up of triangles with a minimum angle of 30° and the distorted grid is composed of triangles stretched in a diagonal direction with an edge smaller than the other two.

Grids used for computations are shown in Figs. 3(a) and 3(b), which are referred to as grid A and B, respectively. Grid A is
made up of 1,880 elements and 1,000 nodes and grid B 1,831 elements and 1,000 nodes. Computations were carried out by using a first-order and second-order scheme with the “minmod,” the “superbee” limiters (Anastasiou and Chan 1997), and the present model. The model was run until a steady state was reached.

Figs. 4(a) and 4(b) show the water surface profiles computed by using the present model and the “superbee” limiter on grid A. Figs. 5(a) and 5(b) show the computed water surface profiles with the present model and “superbee” limiter on grid B. Figs. 4 and 5 indicate that spurious oscillations clearly appear when the “superbee” limiter is used. Fig. 6 shows the contour plot of the computed flow depth on grid A using the present model. The computed angle of the wave front is 29° and the flow depth and the resultant velocity across the jump are 1.554 m and 8.342 m/s, respectively. These values agree well with the analytical solutions β=29.36°, h\text{exact}=1.5543 m, and |u|\text{exact}=8.3421 m/s.

The maximum and minimum values of the water depth and the $L_2$ norm of the error for grids A and B are compared in Table 1, where the $L_2$ norm of the error is defined as

$$L_2 = \sqrt{\frac{\sum_{i=1}^{N} (h_{\text{computed},i} - h_{\text{analytical},i})^2}{\sum_{i=1}^{N} (h_{\text{analytical},i})^2}}$$

in which $i$=cell number and $N$=total number of elements.

As expected, the first-order model produces the smallest numerical oscillations and the largest $L_2$ norm for both grids. The $L_2$ error of the “superbee” limiter for grid A is smallest among the schemes, but the amount of numerical oscillations is largest. As reported by Anastasiou and Chan (1997), the error of the “superbee” limiter is less than that of the “minmod” limiter, but more numerical oscillations are produced. The present model generates the second smallest $L_2$ error, which differs little from the error of the “superbee” limiter, and the smallest numerical oscillations among second-order schemes.

The computed results for grid B clearly show that the reconstruction and limiting is affected by the connectivity of the grid. In contrast to the present model, the “superbee” and “minmod” limiters produce a large amount of numerical oscillations compared with the result of grid A. The “superbee” limiter, which yielded the smallest $L_2$ error for grid A, produced the largest $L_2$ error among the second-order schemes for grid B. Generation of a large amount of numerical oscillations and a loss of accuracy is due to the poor connectivity of grids. However, the present model does not suffer from a loss of accuracy on distorted grids and shows the most accurate results—the smallest $L_2$ error and numerical oscillations—among the second-order schemes. Numerical tests on two different grids indicate that the present model preserves accuracy even on a distorted grid. Since the quality of the generated mesh cannot always be satisfactory, preserving accuracy and stability independent of the shape of grids is a very important feature for the numerical model simulating real flows.

**Dam-Break Flow in a Converging-Diverging Channel with Initially Dry Bed**

Bellos et al. (1992) performed dam break experiments on various flow conditions. One of the experimental test cases of Bellos et al. (1992) is considered here. This test case can validate the numerical model for real fluid flows because it includes some difficulties such as an irregular flow domain and initially dry bed. The ex-
The experimental flume shown in Fig. 7 had a rectangular cross section with variable width. A movable gate located at \( x = 5.0 \) initially separated upstream and downstream regions and was then removed to simulate dam-break. Depths were measured by eight probes located along the center-line of the channel.

The bottom slope of the experimental flume was 0.006 and Manning’s roughness coefficient was 0.012. The initial water level of the upstream region was 0.3 m above the bottom level at \( x = 0 \) and the initial water depth of the downstream region was zero. A free-slip condition was applied at the upstream end and side walls. At the downstream boundary, a free out-fall condition was applied. The flow domain was triangulated with 3,855 elements. Figs. 8(a–c) show the measured and computed flow depths at the three positions shown in Fig. 7. The results are compared with those of Bellos et al. (1991). Bellos et al. (1991) used the MacCormack scheme to solve two-dimensional shallow water equations.

Fig. 8(a) \((x = -8.5 \text{ m})\) indicates that the results of Bellos et al. (1991) slightly underestimate the flow depth; on the other hand, the present model reproduces the measured depths quite well. Fig. 8(b) \((x = -4.0 \text{ m})\) shows that the results of the two models agree well with measured data. Both models show satisfactory results in the reservoir. In the downstream of the dam, the arrival time of the wavefront is accurately predicted in both models. The present model yields quite satisfactory results for the flow depth, while the results by Bellos et al. (1991) slightly overestimate it [Fig. 8(c)].

**Application to Malpasset Dam Break**

The present model is applied to simulate the Malpasset dam break. Malpasset dam was built for irrigation and storage of drinking water and was located in a narrow gorge of the Reyran valley, approximately 12 km upstream of Frejus which is a resort community on the French Riviera. The dam was a double-curvature arch dam with a height of 66.5 m and a crest length of 223 m. The maximum reservoir capacity was \( 5.5 \times 10^7 \text{ m}^3 \). The dam failed in December 1959 due to the rapid rise of the water level.

![Fig. 5. Computed water surface profile on grid B: (a) present model and (b) “superbee” limiter.](image)

![Fig. 6. Contour plot of the computed flow depth on grid A.](image)

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Grid A</th>
<th>Grid B</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( H_{\max} )</td>
<td>( H_{\min} )</td>
</tr>
<tr>
<td>Present model</td>
<td>1.557</td>
<td>0.995</td>
</tr>
<tr>
<td>First-order scheme</td>
<td>1.554</td>
<td>1.000</td>
</tr>
<tr>
<td>Minmod limiter</td>
<td>1.570</td>
<td>1.000</td>
</tr>
<tr>
<td>Superbee limiter</td>
<td>1.619</td>
<td>1.000</td>
</tr>
<tr>
<td>Exact solution</td>
<td>1.554</td>
<td>1.000</td>
</tr>
</tbody>
</table>
level in the reservoir caused by exceptional rainfall. 421 casualties were reported and a large portion of the Esterel freeway was flooded away.

Field and laboratory data about the flood wave due to the Malpasset dam break are available from Electricité de France (EDF). Three electric transformers were destroyed by the flood wave and the exact times of the shutdown are known. The three transformers are denoted as A, B, and C. After the accident, nearly 100 points were surveyed along the left and the right banks of the Reyran river valley by the local police. The observed high water marks at these points can be used to estimate the maximum water level during the flood event. The measured high water marks of 17 significant points among these points, denoted as P1–P17, are available from EDF.

A nondistorted 1/400 scale model was built by Laboratoire National d’Hydraulique (LNH) of EDF in 1964. Fourteen gauges, denoted as S1–S14, were placed in the physical model and the maximum water levels at these gauges during the simulation were measured. The Strickler coefficient was estimated to be in the range of 30–40 m m$^{-1/3}$ s$^{-1}$ (Hervouet and Petitijean 1999). Gauge measurements were in good agreement with the observed high-water marks when these data were rescaled to the real size.

Fig. 9 shows the topography of the computational domain and the locations of surveyed points, gauges, and electronic transformers. Fig. 10 shows the triangular meshes used for computation. Triangles are more densely organized near the dam and along the river. The numbers of nodes and elements of the meshes

---

**Fig. 7.** Geometry of the converging-diverging channel (Bellos et al. 1992)

---

**Fig. 8.** Comparisons of measured and computed hydrographs: (a) measured and computed hydrographs at $x = -8.5$ m; (b) measured and computed hydrographs at $x = -4.0$ m; and (c) measured and computed hydrographs at $x = 10.0$ m

---

**Fig. 9.** Topography and the locations of surveyed points, gauges, and electronic transformers

---

**Fig. 10.** Meshes of upstream regions
used for the computation are 34,849 and 67,719, respectively. The bottom elevation of the valley was obtained from the 1:20,000 ancient IGN (Institut Géographique National) maps. The elevations range from −20 m below sea level to 100 m above sea level. The number of digitized points amounted to 13,541 and the bottom elevation at each node was interpolated from these points.

The dam is considered as a straight line between the points of (x, y) coordinates being (4,701.18 m, 4,143.41 m) and (4,656.5 m, 4,392.10 m), and instantaneous removal of the entire dam at the time \( t = 0 \) is assumed. The initial water level in the reservoir is 100 m above sea level which is equal to zero. In the downstream of the dam, the bottom is set to dry because the discharge through the outlet gate is unknown and its quantity is considered to be small compared with that of the flood wave. The water depth in the sea is also removed for convenience since it does not affect the solution at surveyed points and gauges during computation. A solid boundary condition is imposed along all boundaries and the Manning roughness coefficient is set to 0.033, which equals the Strickler coefficient of 30 m\(^{1/3}\) s\(^{-1}\).

The computation was run until the time \( t = 3,000 \) s. The model was run on a PC with a Pentium 4 2.4 Ghz processor and the run time of the test case was about 7 h 15 min. The run time is considered not excessive given that the number of elements is approximately 70,000.

Fig. 11 presents the computed water depth and velocity distributions near the dam at time \( t = 300 \) s. Fig. 12 shows the computed flood inundation map after 3,000 s and the location of the surveyed points, where the high-water marks were surveyed.

Fig. 11. Computed water depth and velocity distributions at time \( t = 300 \) s

Cells with a water depth smaller than 0.05 m are considered dry. In Fig. 12, the boundaries of the flooded area are generally in good agreement with police surveys.

Two published simulation results of the Malpasset dam break were used for the comparison with the results of the present model. Hervouet and Petitijean (1999) applied TELEMAC-2D to the Malpasset dam break. TELEMAC-2D is commercial software based on a two-dimensional finite element model, which was developed by the Laboratoire Nationale d’Hydraulique from EDF. Valiani et al. (2002) also simulated the Malpasset dam break by the two-dimensional finite volume method on the quadrilateral meshes.

Table 2 shows the measured and computed arrival times of the flood wave to the three electronic transformers and the travel time between A and B, and B and C. The arrival time is assumed to be equal to the wavefront arrival time. Since the exact solution is unknown in actual flows, not only the arrival times of the flood wave but also the travel times of the flood wave between two points are important criteria for judging whether the model reproduces the flow phenomena accurately or not. The results are compared with those of Hervouet and Petitijean (1999) and Valiani et al. (2002). The computed arrival times at A, B, and C of the three models are generally in good agreement with the field data. The present model produces improved results compared with the other two models for the travel time of the wave between A and B, and B and C.

Table 3. Maximum Water Levels at P1–P17 (unit: m)

<table>
<thead>
<tr>
<th>Points</th>
<th>Field data (m)</th>
<th>Present model (m)</th>
<th>Valiani et al. (2002) (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>79.15</td>
<td>75.13</td>
<td>75.96</td>
</tr>
<tr>
<td>P2</td>
<td>87.20</td>
<td>87.38</td>
<td>89.34</td>
</tr>
<tr>
<td>P3</td>
<td>54.90</td>
<td>55.09</td>
<td>53.77</td>
</tr>
<tr>
<td>P4</td>
<td>64.70</td>
<td>57.41</td>
<td>59.64</td>
</tr>
<tr>
<td>P5</td>
<td>51.10</td>
<td>47.11</td>
<td>45.56</td>
</tr>
<tr>
<td>P6</td>
<td>43.75</td>
<td>45.74</td>
<td>44.85</td>
</tr>
<tr>
<td>P7</td>
<td>44.35</td>
<td>40.47</td>
<td>42.86</td>
</tr>
<tr>
<td>P8</td>
<td>38.60</td>
<td>32.58</td>
<td>34.61</td>
</tr>
<tr>
<td>P9</td>
<td>31.90</td>
<td>33.16</td>
<td>32.44</td>
</tr>
<tr>
<td>P10</td>
<td>40.75</td>
<td>38.29</td>
<td>38.12</td>
</tr>
<tr>
<td>P11</td>
<td>24.15</td>
<td>25.16</td>
<td>25.37</td>
</tr>
<tr>
<td>P12</td>
<td>24.90</td>
<td>25.96</td>
<td>27.35</td>
</tr>
<tr>
<td>P13</td>
<td>17.25</td>
<td>24.41</td>
<td>23.58</td>
</tr>
<tr>
<td>P14</td>
<td>20.70</td>
<td>20.58</td>
<td>23.19</td>
</tr>
<tr>
<td>P15</td>
<td>18.60</td>
<td>19.08</td>
<td>19.37</td>
</tr>
<tr>
<td>P16</td>
<td>17.25</td>
<td>17.04</td>
<td>20.39</td>
</tr>
<tr>
<td>P17</td>
<td>14.00</td>
<td>16.00</td>
<td>14.23</td>
</tr>
</tbody>
</table>
Table 3 and Fig. 13 show the comparison of numerical results of the present model and Valiani et al.’s model with field data for the maximum water levels at surveyed points. Good agreement is observed in both models.

In Table 4 and Fig. 14, the experimental data of the maximum water level and the numerical results are compared. Only the gauges outside the reservoir (S6–S14) are included. The results of the three models are quite similar.

Conclusions

A numerical model based on the finite volume method has been developed and tested to simulate shallow water flows over a wet or dry bed with complicated boundaries.

Advantages of the present model are the capabilities of handling complex geometry by using unstructured grids and solving discontinuous solution stably by approximate Riemann solver. The major advantage of the model is to use a multidimensional reconstruction technique and a continuously differentiable multidimensional limiter for obtaining a second-order spatial accuracy. It enables the model to maintain accuracy and stability even on highly distorted grids independent of grid connectivity.

The oblique hydraulic jump tests on well-connected and distorted grids show that accuracy and stability of the present model are preserved even on highly distorted grids. In the dam break test of Bellos et al. (1992), the computed depths showed good agreement with experimental data and produced improved results compared with those of Bellos et al. (1992). In the Malpasset dam break test, the computed results were in very good agreement with those of other models, and field and experimental data. This proves the applicability of the present model to real flows.

Acknowledgments

The writers are grateful to the anonymous reviewers for their constructive comments on the manuscript and would like to thank Professor C. V. Bellos for providing experimental results of the dam break test.

Notation

The following symbols are used in this paper:

- $A$ = area of a control volume;
- $c$ = wave celerity;
- $E$ = flux vector;
- $F$ = flux vector in the $x$ direction;
- $G$ = flux vector in the $y$ direction;
- $g$ = acceleration due to gravity;
- $g_a$, $g_b$, $g_c$ = functions of gradients of surrounding cells;
- $h$ = flow depth;
- $h_{ab}$ = $h$ at boundary;
- $I$ = identity matrix;
- $i$ = element index;
- $j$ = cell face index;
- $k$ = vertex index;
- $M$ = number of elements surrounding a vertex;
- $N$ = number of total elements;
- $n$ = Manning’s roughness coefficient;
- $n$ = time level;
- $n$ = unit outward vector normal to cell face;
- $R_i$ = nearest distance from centroid of control volume to cell vertices;
- $R^R, R^L$ = Riemann invariants;
- $r$ = position vector;
- $S$ = vector of source terms;
- $S_f$ = vector of friction source terms;
- $S_s$ = vector of slope source terms;
- $S_{fx}$ = friction slope in $x$ direction;
- $S_{fy}$ = friction slope in $y$ direction;
- $S_{sx}$ = bed slope in $x$ direction;
- $S_{sy}$ = bed slope in $y$ direction;
- $t$ = unit tangential vector at the cell face;
\begin{align*}
t &= \text{temporal coordinate;} \\
U &= \text{vector of the conserved variables;} \\
u &= \text{vertically averaged velocities in } x \text{ direction;} \\
u_b &= u \text{ at boundary;} \\
v &= \text{vertically averaged velocities in } y \text{ direction;} \\
v_b &= v \text{ at boundary;} \\
x, y &= \text{orthogonal Cartesian coordinates;} \\
z &= \text{bottom elevation;} \\
\Delta &= \text{boundary of a control volume;} \\
\nabla &= \text{nabla operator;} \\
\omega_a, \omega_b, \omega_c &= \text{weights used to limit gradient of solution.}
\end{align*}

References


